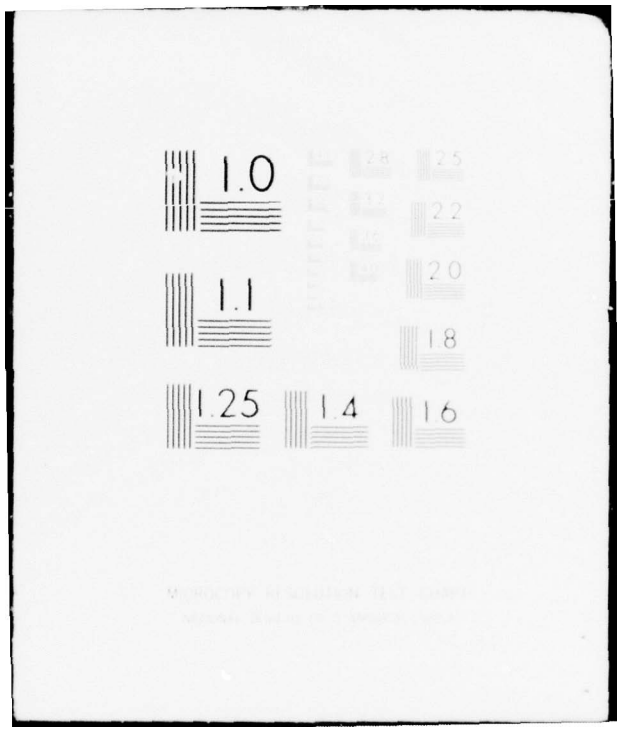
AD-A046 577 RAND CORP SANTA MONICA CALIF F/G 9/2
ESTIMATION TECHNIQUES AND OTHER WORK ON IMAGE CORRELATION, (U)
SEP 77 J A RATKOVIC, F W BLACKWELL F49620-77-C-0023
UNCLASSIFIED R-2211-AF NL

1 OF 1
AD A046577

END
DATE FILMED
12-77
DDC


1.0
1.1
1.25
1.4
1.6
2.2
2.0
1.8

R-2211-AF
September 1977

Estimation Techniques and Other Work on Image Correlation

J. A. Ratkovic, F. W. Blackwell, H. H. Bailey, C. L. Lowery

A Project AIR FORCE report
prepared for the
United States Air Force

Rand
SANTA MONICA, CA. 90406

The research reported here was sponsored by the Directorate of Operational Requirements, Deputy Chief of Staff/Research and Development, Hq. USAF under Contract F49620-77-C-0023. The United States Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon.

Published by The Rand Corporation

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: R-2211-AF	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): Estimation Techniques and Other Work on Image Correlation,	5. TYPE OF REPORT & PERIOD COVERED: Interim 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): J. A./Ratkovic, F. W./Blackwell, H. H./Bailey, C. L./Lowery	8. CONTRACT OR GRANT NUMBER(s): F49620-77-C-0023
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The Rand Corporation, 1700 Main Street, Santa Monica, Ca. 90406	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Project AIR FORCE Office (AF/RDQA), Directorate of Operational Requirements, Hq USAF, Washington, D.C. 20330	12. REPORT DATE: Sep [redacted] 77 13. NUMBER OF PAGES: 78p. 72
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

No restrictions

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Images
Pattern Recognition
Target Discrimination

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

see reverse side

296600

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Image correlation, or "map matching," has important implications for autonomous target acquisition and terminal guidance for various missiles, as well as for other pattern recognition applications. This report makes the following contributions to image correlation: (1) Analysis of block-substitution effects (snow, shadows, clouds, and the like) on the probability of correlation, Pc, (2) Development of a closed-form approximation for computing Pc, (3) New techniques for calculating the inherent scene characteristics (numbers of independent elements in the scene and the scene correlation length), (4) A completely new procedure for estimating the value of Pc from the correlation data themselves, and (5) A new technique for selecting and locating the significant features in a scene. (Author)

R-2211-AF
September 1977

Estimation Techniques and Other Work on Image Correlation

J. A. Ratkovic, F. W. Blackwell, H. H. Bailey, C. L. Lowery

A Project AIR FORCE report
prepared for the
United States Air Force

Rand
SANTA MONICA, CA. 90406

PREFACE

The work described in this report was done under the Project AIR FORCE (formerly Project RAND) study effort entitled "Target Acquisition." The specific subject is "map matching," or image correlation, to achieve autonomous target acquisition and terminal guidance for various missiles. The report updates and supplements two recent Rand publications* on the same subject, and some familiarity on the part of the reader with those publications is presumed. This research should be useful to Air Force and other Department of Defense agencies and their contractors who are concerned with technical aspects of air-to-ground attack, as well as others working in the basic field of image correlation.

---

*H. H. Bailey, F. W. Blackwell, C. L. Lowery, and J. A. Ratkovic, *Image Correlation, Part I: Simulation and Analysis*, R-2057/1-PR, November 1976; and H. W. Wessely, *Image Correlation, Part II: Theoretical Basis*, R-2057/2-PR, November 1976.

SUMMARY

This report makes the following contributions to image correlation:

1. Analysis of block-substitution effects (snow, shadows, clouds, and the like) on the probability of correlation, P_c.
2. Development of a closed-form approximation for computing P_c.
3. New techniques for calculating the inherent scene characteristics (number of independent elements in the scene and the scene correlation length).
4. A completely new procedure for estimating the value of P_c from the correlation data themselves.
5. A new technique for selecting and locating the significant features in a scene.

Block-substitution errors result when portions of the scene to be imaged suffer uniform amplitude errors such as might be produced by shadows in the scene or snow on the ground. This report analyzes the effect of this class of errors and determines the degradation in P_c as a function of the magnitude of the amplitude error and the sizes of the sensor and reference maps involved.

The computation of P_c heretofore has involved numerical integration techniques. The immediate consequence of the approximation technique developed in this report is to simplify the computational work. Additionally, (1) some insight has been obtained into the variables that contribute most significantly to improving P_c; (2) it has become easier to implement the estimation procedure for determining P_c directly from the correlation data; and (3) using this algorithm may enable one to identify a priori the distinguishing features in a scene.

The terminology used most often in describing the statistical characteristics of a scene include one of several definitions for the scene correlation length and the number of independent elements in the scene. The usual one-dimensional calculation is a primary limitation when computing the correlation length for area applications. The new

technique developed in this report works backward from the correlation data to calculate the number of independent elements in the scene--from which an effective correlation length can easily be calculated. The technique was verified using a randomly generated scene. The assumption of Gaussian statistics in these calculations also proved reasonably representative of the statistics of the real terrain imagery tested.

The heart of the report describes a new procedure for estimating the P_c value directly from the data. The technique again works backward from the correlation data to estimate the scene characteristics--i.e., the number of independent elements in the scene, the variance in scene intensity, and the signal-to-noise (S/N) ratio. From these estimates of the scene characteristics, one can calculate the number of independent displacement positions at which the two maps are compared and the variance of the in-register correlation function. With these estimates and the P_c closed-form approximation, one can compute P_c directly.

This technique was tested using Monte Carlo simulations with real imagery. In these simulations, the correct match point was identified in all cases (P_c SIMULATED = 1). Values determined using the estimation technique for each individual correlation experiment varied between 0.83 and 0.99. Perhaps even more importantly, a separate experiment was set up wherein two different maps were correlated. Obviously, for this situation, P_c SIMULATED = 0. The estimation technique in this case produced values no greater than 0.24. Thus, the estimation technique appears to be capable of distinguishing between cases where the sensor map is or is not contained in the reference map.

Finally, a new technique for selecting and locating the significant features in a scene has been developed. The technique is based on estimating the value of P_c for a large number of submaps and selecting those submaps (sets of points constituting the "significant" features) that have the greatest effect on the value of P_c. An exploratory computational experiment showed sufficiently encouraging results to warrant further research on this technique.

CONTENTS

I. REVIEW OF PREVIOUS WORK

This report updates previous Rand publications on image correlation[1,2] and describes some interesting new work. To provide continuity, results of the earlier studies are reviewed in this section.

The first major conclusion was that an approximate lower bound on the probability of correct target acquisition, P_c, can be calculated, so that we can, at least in principle, design systems to meet an acquisition specification.

Quantitative relationships in Refs. 1 and 2 showed the dependence of P_c on N (the sensor map size), M (the search area or reference map size), S/N (nominally the signal-to-noise ratio but, more importantly, a measure of the fidelity of the reference map vis à vis the real-time sensor map), and various parameters describing systematic intensity and geometric errors. Thus, one has the tools for carrying out design tradeoffs on sensor resolution and field of view (to increase N), on midcourse navigation (to decrease M), on attitude reference and guidance (to reduce geometrical distortions), on data processing capabilities (to reduce both synchronization and quantization effects), on more recent and more accurate reference data (to increase the S/N ratio), and so on, including finally a tradeoff of the cost of increasing the P_c requirement itself with the loss of those few weapons that would be wasted if they achieved a false lock-on.

Most of these relationships for P_c were derived from a simple Gaussian theory that is known to be unrealistic. Fortunately, however, this theory appears to err on the conservative side--most scenes are more distinctive than assumed and results are better (i.e., yield fewer gross errors) than predicted. On the other hand, real systems have additional error sources that were not analyzed or simulated. Nevertheless, the important point is that, with a "floor" established for P_c, there should be no major surprises in future flight tests of image-correlation hardware. Improvements in the theory, and additional data from simulation experiments using specific scenes of interest, are expected to improve the predictions and relax some of the design

constraints. One can design to P_c requirements, though at the moment not as effectively as is desired.

The second major conclusion was the following: First, since a review of the theory showed that none of the commonly used algorithms is necessarily optimum (all are approximations to a definable but unattainable ideal), we are free to look for other algorithms and new approaches; and second, since the best way to avoid false locks is to provide a high S/N ratio, and since one way of achieving a high S/N ratio would be to enhance the characteristic features in a scene and throw away the rest of the data (which contributes mostly noise), a deliberate effort to search out the unique or distinctive features in any scene might be highly desirable.

We do not know at this time how to extract the "characteristic features" or invariant properties of a scene--those least likely to change with time or environmental conditions and therefore most likely (in this context) to lead to high values of the signal-to-noise ratio. However, the discussion of error sources in Sec. III of Ref. 1 made it clear that the useful information resides more in the geometric relationships than in the intensity or signal amplitude dimension of an image. Absolute intensity levels are the least dependable quantities, and intensity ratios and even the algebraic sign of differences (contrast) or gradients are unreliable. But the locations of most intensity boundaries are fixed, subject only to certain geometrical distortions that are (a) limited in magnitude (largely controlled by the system design) and (b) constant or slowly varying over a given image.

In particular, it was suggested that techniques currently being developed in the field of pattern recognition should be explored and exploited for this purpose. These might even include some of the complex, heuristic, upward and downward directed, hypothesis-testing techniques used in so-called artificial intelligence programs. But also, at least for the present, one should not rule out completely the use of people to examine the reconnaissance imagery and pick what "look like" good features, followed by the selection on a specific ad hoc basis of "filters" or special image-processing algorithms appropriate to each

scene. In any case, the on-board processing would then consist of applying only the selected filters, looking solely for those features known to be significant in that weapon's assigned target area, followed (presumably) by a simple correlation or pseudo-correlation algorithm for locating the match point. It is anticipated that in this way algorithms more efficient than those in practice to date will evolve, and that at the same time higher values of P_c will result. To sum up, the suggestion was made (independently of similar prior suggestions by others) that emphasis might well be shifted away from schemes that compare every picture element (pixel) in the reference and sensor images to searches for ways to extract the invariant and "distinctive" features in any given scene.

It must be pointed out that, despite the strength of the foregoing suggestion for a change in research direction, it has not been demonstrated that the feature-extraction route is the "only" way to go. For the foreseeable future it will be expensive, and ad hoc solutions may have to be found for almost every scene; and there may be many scenes and conditions for which one or more of the conventional correlation systems will suffice. For the time being, probably both courses should be pursued; but we *think* that, in order to achieve a multi-spectral capability, the future lies in feature extraction. It may lie even further in the direction of pattern recognition per se, as differentiated from conventional correlation with images reduced through feature extraction; but clearly that choice cannot be made at this time.

II. ANALYSIS OF BLOCK-SUBSTITUTION ERRORS

Some preliminary "block substitution" simulation experiments were described very briefly in Sec. III (p. 42) of Ref. 1. The supporting analyses were carried out too late to be included in that report but are presented here.

Whenever contiguous areas suffer uniform amplitude errors, these errors are referred to as block substitutions. Shadows due to scattered low clouds or changes in sun angle can cause dark blocks, and intervening sunlit clouds and certain kinds of jamming can produce bright blocks. If only one type of error is treated at a time, such blocks can be characterized by just two parameters--α, the average (constant) amplitude of the signal in the blocks, and β, the fraction of the total area that is so affected.

In general, these errors have two effects on the image correlation. First, only that portion of the sensor map which is not changed contributes to the correlation peak. Hence, the absolute value of the difference between the in-register and out-of-register values of the correlation function should be reduced by the factor $1 - \beta$. Second, the pixels that are changed effectively add a significant amount of "noise" into the correlation. These effects can be analyzed quantitatively, following the general procedures described in Ref. 1, by calculating the expected value and the variance for the in- and out-of-register correlation function as functions of α and β, and then computing P_c by integrating the following expression [identical to Eq. (4) of Ref. 1]

$$P_c = \frac{1}{\sqrt{2\pi}\,\sigma_c} \int_{-\infty}^{+\infty} \exp\left[\frac{-w^2}{2\sigma_c^2}\right] \cdot \left[\frac{1}{2} + \frac{1}{2}\,\mathrm{erf}\,\frac{(\bar{\phi}_o - \bar{\phi}_j + w}{\sqrt{2}\,\sigma_j}\right]^{O} dw \qquad (1)$$

where w is $\phi_o - \bar{\phi}_o$, erf is the error function (see footnote on p. 14), O is the number of possible out-of-register positions of the sensor map,

and the statistical quantities ($\bar{\phi}_o$, $\bar{\phi}_J$, σ_o, and σ_J) can be shown to have the values given in Tables 1 and 2 for the Product and MAD (mean absolute difference) algorithms, respectively.

A rather large number of numerical cases have been evaluated. Two values of α were used: zero (with no noise added, possibly simulating a shadow) and $3\sigma_x$ (roughly the maximum signal level, with various amounts of noise added, to represent the second category). Several values of β have been used, typically 0.1, 0.3, 0.5, and up to 0.7. Typical results are summarized in Figs. 1 through 4. Figures 1 and 2 show the dependence of P_c on both α and β, and Figs. 3 and 4 show more clearly the β-dependence. As would be expected, increasing β decreases P_c, and the effects are always worse when α deviates from zero (the mean value of the signal).

Several specific cases have been extracted and listed in Tables 3 through 6 to illustrate the following points. With the MAD algorithm, the P_c achieved when block substitutions are present is always less than if the unchanged sensor elements, $N(1 - \beta)$ in number, were the only elements affecting the correlation process; that is to say, the additional noise introduced still further degrades P_c. For the Product algorithm, the changed blocks do not contribute anything additional to the noise when the intensity level is equal to the mean ($\alpha = 0$); in that case the reduction in the number of operative pixels is the only effect. However, as α deviates from the mean, the additional noise effects are again evident in the last column of Table 6. Finally, for low S/N ratios (i.e., high noise levels), this additive noise effect is not very significant; but as S/N increases, especially when S/N >> 1, the added noise decreases P_c markedly, as seen particularly from the last entries in Tables 3 and 4.

For completeness, the simulation results from Ref. 1 are repeated here in Table 7. As expected and as predicted by the analysis, degradation increases with increasing size of the substituted blocks and with deviation of the substituted value from the signal mean. It also depends somewhat on scene type. Finally, the Normalized Product algorithm (NProd) is much more resistant to large-block substitution errors than is the MAD algorithm, particularly in the case of shadows; but both are seriously degraded in the presence of large amounts of jamming.

Table 1

RELEVANT ENSEMBLE STATISTICS FOR PRODUCT ALGORITHM

$$\bar{\Phi}_o = (1 - \beta)\sigma_x^2$$

$$N\sigma_o^2 = (1 - \beta)\left(2\sigma_x^4 + \sigma_x^2\sigma_n^2\right) + \beta A^2\sigma_x^2$$

$$\bar{\Phi}_j = 0$$

$$N\sigma_j^2 = (1 - \beta)\left(\sigma_x^4 + \sigma_x^2\sigma_n^2\right) + \beta A^2\sigma_x^2$$

Table 2

RELEVANT ENSEMBLE STATISTICS FOR MAD ALGORITHM

$$\bar{\Phi}_o = \sqrt{\frac{2}{\pi}}\,(1 - \beta)\sigma_n + 2\beta\left(\alpha\ \mathrm{erf}\ \frac{\alpha}{\sigma_x} + \frac{\sigma_x}{\sqrt{2\pi}}\ \exp\ \frac{-\alpha^2}{2\sigma_x^2}\right)$$

$$N\sigma_o^2 = \left(1 - \frac{2}{\pi}\right)(1 - \beta)\sigma_n^2 + \beta\left\{\left(\sigma_x^2 + \alpha^2\right) - 4\left(\alpha\ \mathrm{erf}\ \frac{\alpha}{\sigma_x} + \frac{\sigma_x}{\sqrt{2\pi}}\ \exp\ \frac{-\alpha^2}{2\sigma_x^2}\right)\right\}$$

$$\bar{\Phi}_j = \sqrt{\frac{2}{\pi}}\,(1 - \beta)\sqrt{2\sigma_x^2 + \sigma_n^2} + 2\beta\left(\alpha\ \mathrm{erf}\ \frac{\alpha}{\sigma_x} + \frac{\sigma_x}{\sqrt{2\pi}}\ \exp\ \frac{-\alpha^2}{2\sigma_x^2}\right)$$

$$N\sigma_j^2 = \left(1 - \frac{2}{\pi}\right)(1 - \beta)\left(2\sigma_x^2 + \sigma_n^2\right) + \beta\left\{\left(\sigma_x^2 + \alpha^2\right) - 4\left(\alpha\ \mathrm{erf}\ \frac{\alpha}{\sigma_x} + \frac{\sigma_x}{\sqrt{2\pi}}\ \exp\ \frac{-\alpha^2}{2\sigma_x^2}\right)\right\}$$

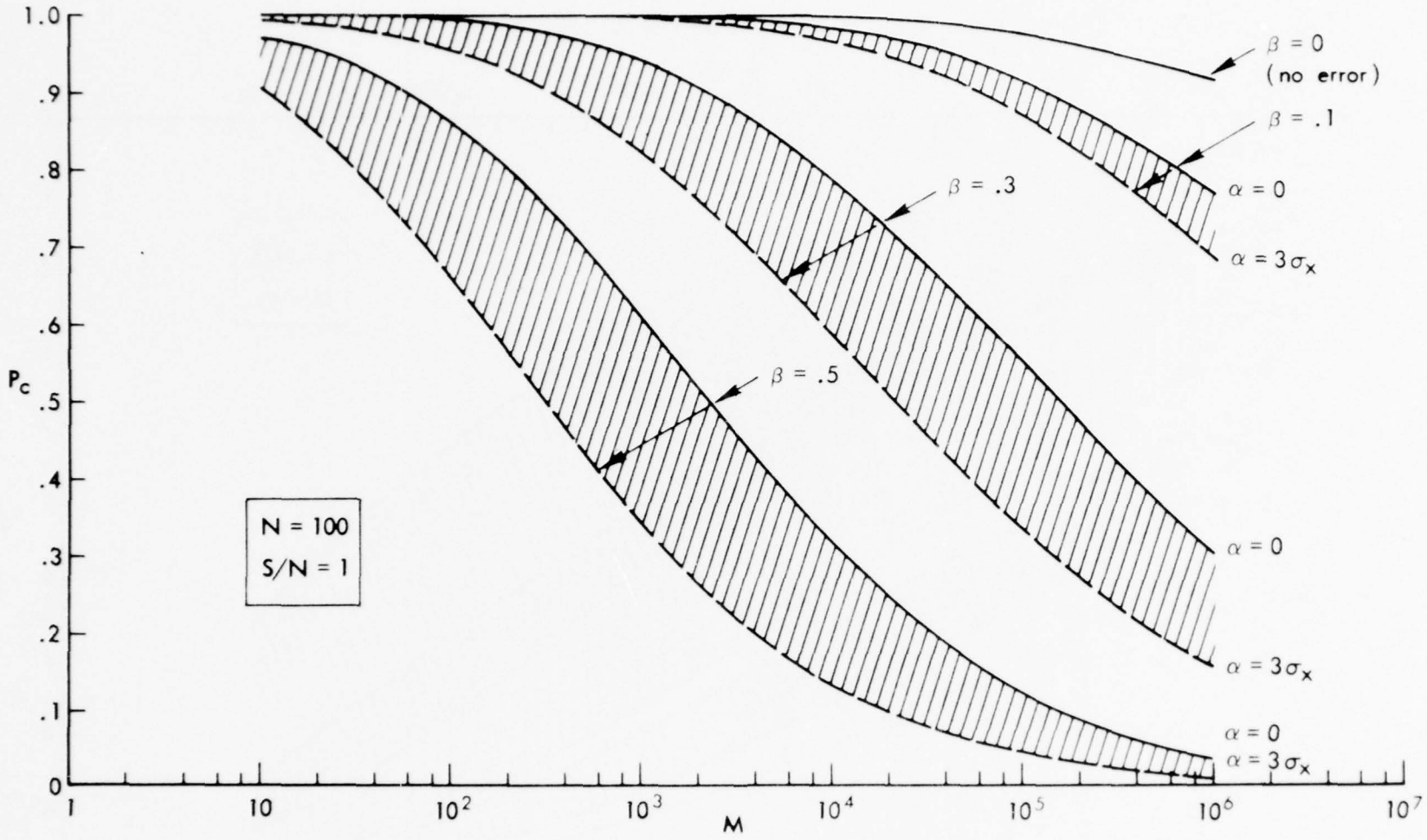


Fig. 1 — The effects on P_c of a variation in α:

MAD algorithm

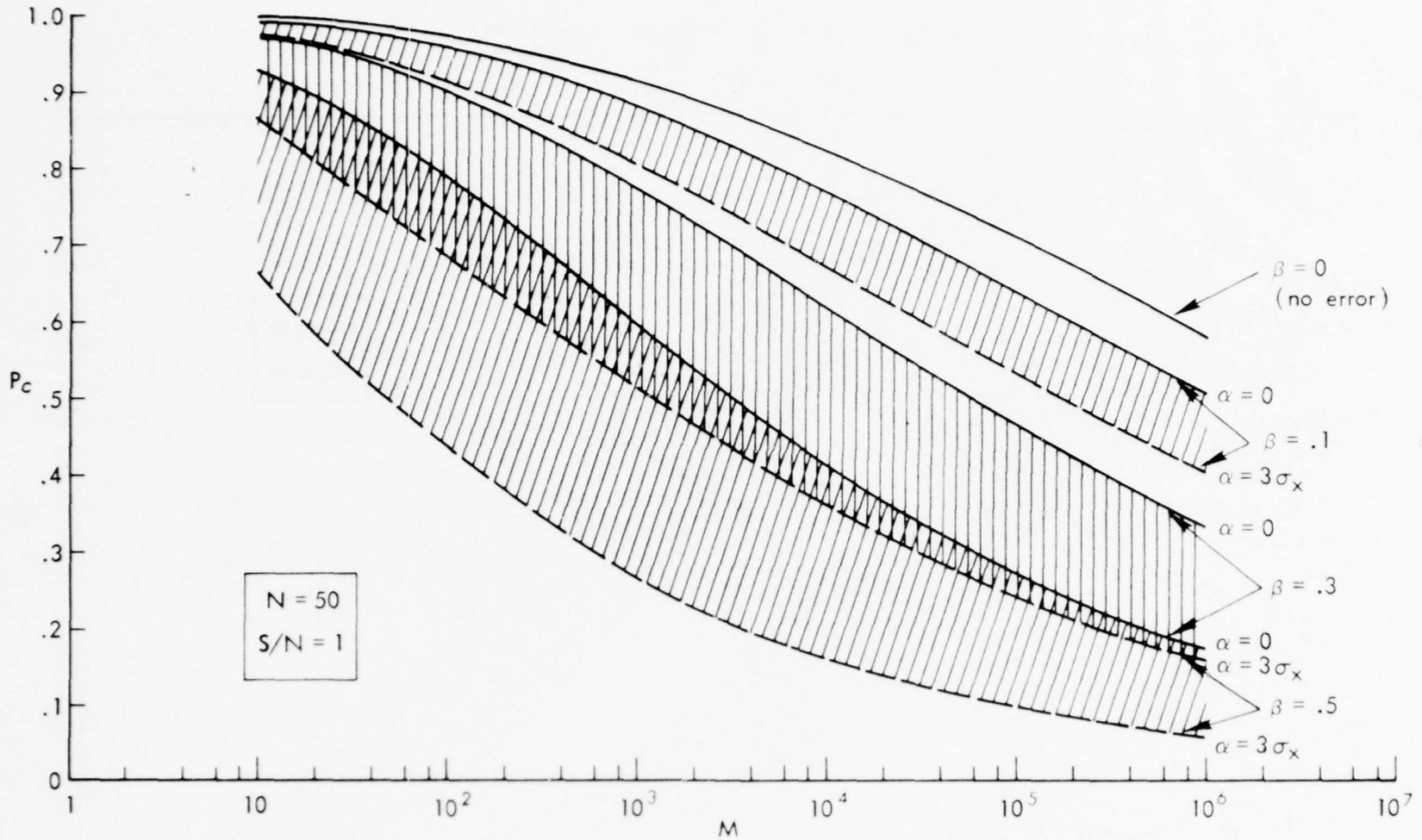


Fig. 2 — The effects on P_c of a variation in α:

Product algorithm

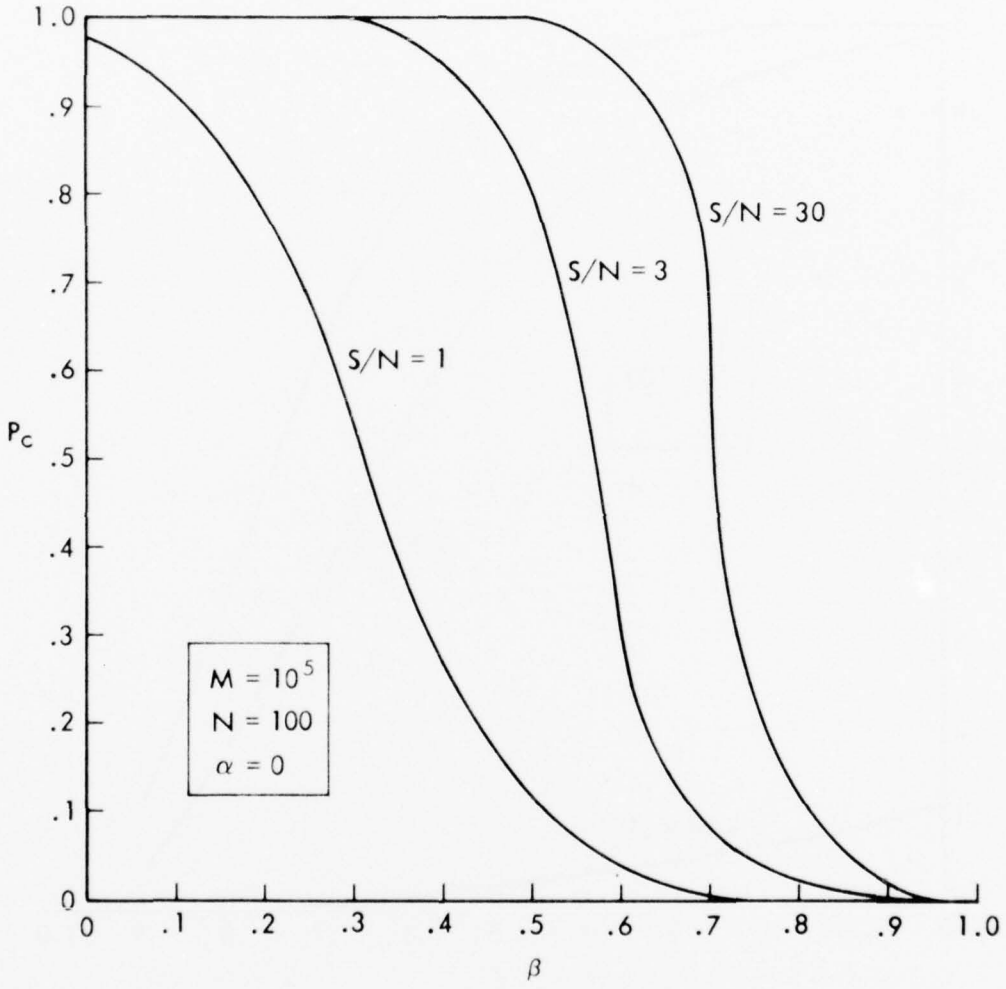


Fig. 3 — The effects on P_c of a variation in β: MAD algorithm

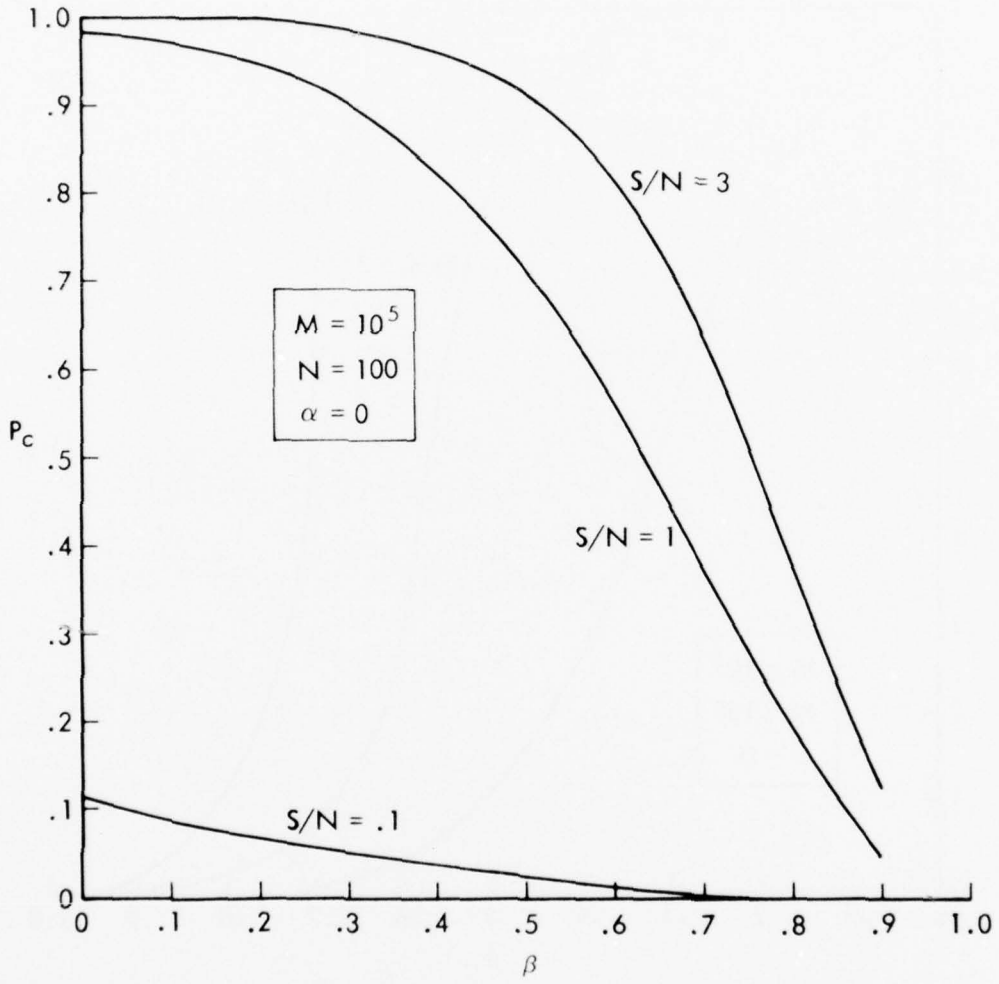


Fig. 4 — The effects on P_c of a variation in β: Product algorithm

Table 3

EFFECTS OF BLOCK SUBSTITUTIONS ON P_c:
MAD ALGORITHM, $\alpha = 0$

S/N	N	β	Q	P_c	ΔP_c
0.1	2000	0.5	10^3	0.62	0.05
0.1	1000	0	10^3	0.67	
1	200	0.5	10^5	0.77	0.19
1	100	0	10^5	0.98	
3	50	0.3	10^5	0.62	0.28
3	35	0	10^5	0.90	
30	30	0.5	10^5	0.15	0.60
30	15	0	10^5	0.75	

Table 4

EFFECTS OF BLOCK SUBSTITUTIONS ON P_c:
MAD ALGORITHM, $\alpha = 3\sigma$

S/N	N	β	Q	P_c	ΔP_c
0.1	3000	0.5	10^3	0.79	0.11
0.1	1500	0	10^3	0.90	
1	1000	0.7	10^5	0.90	0.10
1	300	0	10^5	1.0	
30	30	0.5	10^5	0.75	0.71
30	15	0	10^5	0.04	

Table 5

EFFECTS OF BLOCK SUBSTITUTIONS ON P_c:
PRODUCT ALGORITHM, $\alpha = 0$

S/N	N	β	Q	P_c	ΔP_c
0.1	500	0.5	10^4	0.81	0
0.1	250	0	10^4	0.81	
1	100	0.5	10^5	0.71	0
1	50	0	10^5	0.71	
3	100	0.7	10^5	0.63	0
3	30	0	10^5	0.63	
30	100	0.5	10^6	0.94	0
30	50	0	10^6	0.94	

Table 6

EFFECTS OF BLOCK SUBSTITUTIONS ON P_c:
PRODUCT ALGORITHM, $\alpha = 3\sigma$

S/N	N	β	Q	P_c	ΔP_c
0.1	1000	0.5	10^5	0.73	0.26
0.1	500	0	10^5	0.99	
1	100	0.5	10^5	0.23	0.48
1	50	0	10^5	0.71	
30	100	0.5	10^5	0.63	0.34
30	50	0	10^5	0.97	

Table 7

EFFECT ON CERTAIN BLOCK SUBSTITUTION ERRORS ON THE PROBABILITY OF CORRELATION

(Noise added to sensor scene, S/N = 3, N = 100, Q = 120)

"Shadow" Effect, $\alpha = 0$

	Fraction of Sensor Blocked Out, β					
	0.3		0.5		0.7	
Scene Type	MAD	NProd	MAD	NProd	MAD	NProd
Agricultural	1.00	1.00	.64	.84	.12	.56
Mountain	.96	1.00	.80	.84	.32	.68
Desert	1.00	1.00	.96	1.00	.36	.84
Suburban	1.00	1.00	.76	.96	.48	.68

"Jamming" Effect, $\alpha = 3\sigma$

	Fraction of Sensor Blocked Out, β					
	0.3		0.5		0.7	
Scene Type	MAD	NProd	MAD	NProd	MAD	NProd
Agricultural	.64	.68	.24	.36	.04	.04
Mountain	.92	.88	.32	.28	.00	.08
Desert	.80	.72	.32	.40	.00	.24
Suburban	.64	.52	.20	.36	.00	.08

III. A CLOSED-FORM APPROXIMATION FOR P_c

The most important new development in the Rand study following publication of Refs. 1 and 2 has been the generation of an approximate expression for computing P_c in closed form. The immediate consequence of this approximation is to simplify the computational effort in obtaining P_c. Additionally, there have been three other consequences. First, we have gained some insight into the variables that contribute most significantly to improving P_c, as described below. Second, with this simplified closed-form expression, it becomes a much easier task to estimate the value of P_c, in real time, directly from the correlation data. This latter point is discussed in the next section. Third, the use of P_c may lead to a method for identifying a priori the most distinguishing features in a scene. This is discussed in Sec. V of this report.

It has been shown[1] that the integral that must be evaluated (by numerical techniques) in determining P_c for correlation algorithms is of the general form:

$$P_c = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-Z^2}{2}\right) \left[1/2 \pm \operatorname{erf}(AZ + B)\right]^Q dZ \quad \begin{pmatrix}\max\\ \min\end{pmatrix} \text{algorithm}^* \tag{2}$$

*For a maximizing algorithm (e.g., Product) the positive sign in front of the erf is used and for a minimizing algorithm (e.g., MAD) the negative sign is used. Note also that throughout this section the error function is defined as:

$$\operatorname{erf}(Z) = \frac{1}{\sqrt{2\pi}} \int_0^Z e^{-x^2/2}\, dx$$

whereas in Ref. 1 and in Sec. II of this report, it is defined as

$$\operatorname{erf}(Z) = \frac{2}{\sqrt{\pi}} \int_0^Z e^{-x^2}\, dx$$

where $A = \frac{\sigma(0)}{\sigma(J)}$

$B = \frac{E\{\phi(0)\} - E\{\phi(J)\}}{\sigma(J)}$

Q = total number of displacement positions (excluding the match position) at which the sensor and reference map are compared

$E\{\phi(0)\}$, $E\{\phi(J)\}$, $\sigma(0)$, $\sigma(J)$ = ensemble statistics (dependent on the algorithm choice and the S/N ratio).

An approximation technique, taken from Ref. 3, has been suggested by H. W. Wessely of Rand. The essence of this technique is to approximate the quantity

$$F(Z) = [1/2 + \text{erf}\ (AZ + B)]^{Q} \tag{3}$$

contained in Eq. (2) by a step function. Thus, this function (for a maximizing algorithm) will be replaced by

$$F(Z) = \begin{cases} 1 & \text{for} \quad Z \geq Z_o \\ 0 & \text{for} \quad Z < Z_o \end{cases}$$

The rationale for this approximation is that for large values of Q and A (which are normal in the cases of greatest interest), F(Z) transitions between its extreme values of zero and one fairly rapidly in a continuous, nondecreasing manner. The transition point, Z_o, can be found by setting $F(Z_o)$ equal to a half and solving for Z_o. Thus,

$$Z_o = \frac{(+)\ \text{erf}^{-1}\left[(1/2)^{1/Q} - (1/2)\right] - B}{A} \quad \begin{pmatrix} \text{max} \\ \text{min} \end{pmatrix} \text{algorithm} \tag{4}$$

Defining

$$K = \text{erf}^{-1}\left[(1/2)^{1/Q} - (1/2)\right] \tag{5}$$

and substituting into Eq. (4) yields

$$Z_0 = \frac{(+)\ K - B}{A} \qquad \left(\begin{matrix}\max \\ \min\end{matrix} \text{ algorithm}\right) \tag{6}$$

The quite intractable integral in Eq. (2) is thus reduced to

$$P_c = \int_{Z_0}^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-Z^2}{2}\right) dZ \qquad (\text{max algorithm})^* \tag{7}$$

which is readily evaluated to yield

$$P_c = 1/2\ (\overline{+})\ \text{erf}\left[\frac{(+)\ K - B}{A}\right] \qquad \left(\begin{matrix}\max \\ \min\end{matrix} \text{ algorithm}\right) \tag{8}$$

Table 8 compares the approximate values obtained for P_c [using Eq. (8)] with the exact values [using Eq. (2)] for the MAD and Product algorithms. The approximation looks good for values of Q as low as 100, which is typically the minimum number that would be used in any correlation process. Table 9 lists the values of A and B' ($B' = B/\sqrt{N}$) obtained for these algorithms as a function of the S/N ratio. As can be seen from Eq. (3) and plots thereof, the larger the value of A (where A^2 is the ratio of the in- and out-of-register variances), the better is the step function approximation for F(Z). The greatest deviations between exact and approximate values of P_c do not appear to result from using small values of Q so much as from small values of A.

Larger deviations between exact and approximate values for P_c can be seen in Table 8 for the MAD algorithm with S/N ratios for which A is less than 0.5 (S/N = 10 and S/N = 30). However, even for values of A as low as computed in Table 9 (A = 0.13), the approximation is still not too bad. Since A is computed in the process of approximating P_c by Eq. (8), it can also serve as an indicator of how well the

*Limits $-\infty$ to Z_o for a min algorithm.

Table 8

COMPARISON OF EXACT AND APPROXIMATE VALUES FOR P_c

S/N	N	Q	P_c for Product		P_c for MAD	
			Approximate	Exact	Approximate	Exact
0.1	30	10^2	0.22	0.23	0.02	0.1
0.1	30	10^3	0.07	0.07	0.00	0.00
0.1	30	10^4	0.03	0.03	0.00	0.00
0.1	100	10^2	0.70	0.68	0.08	0.09
0.1	100	10^3	0.44	0.42	0.01	0.01
0.1	100	10^4	0.26	0.23	0.00	0.00
0.1	300	10^2	1.0	0.99	0.31	0.31
0.1	300	10^3	0.98	0.96	0.11	0.10
0.1	300	10^4	0.93	0.90	0.03	0.03
0.1	1000	10^2	1.00	1.00	0.90	0.87
0.1	1000	10^3	1.00	1.00	0.70	0.67
0.1	1000	10^4	1.00	1.00	0.49	0.44
1	10	10^2	0.43	0.42	0.12	0.15
1	10	10^3	0.22	0.22	0.01	0.01
1	10	10^4	0.12	0.11	0.00	0.00
1	30	10^2	0.88	0.86	0.85	0.79
1	30	10^3	0.71	0.69	0.42	0.41
1	30	10^4	0.56	0.52	0.14	0.13
1	80	10^2	1.00	1.00	1.00	1.00
1	80	10^3	0.99	0.99	1.00	1.00
1	80	10^4	0.98	0.98	0.99	0.97
10	10	10^2	0.66	0.64	1.00	0.94
10	10	10^3	0.45	0.44	0.66	0.58
10	10	10^4	0.32	0.30	0.03	0.05
10	30	10^2	0.98	0.97	1.00	1.00
10	30	10^3	0.93	0.93	1.00	1.00
10	30	10^4	0.87	0.84	1.00	1.00
30	10	10^2	0.68	0.66	1.00	0.99
30	10	10^3	0.48	0.47	1.00	0.88
30	10	10^4	0.34	0.32	0.41	0.33

approximation will match the exact value. As a rule of thumb, the data indicate that values of A probably as low as 0.10 would yield reasonable approximations to the exact value of P_c.

Table 9

VALUES OF A AND B' FOR THE MAD AND PRODUCT ALGORITHMS

	Product		MAD	
S/N	A	B'	A	B'
0.1	1.04	0.30	0.90	-0.115
1.0	1.22	0.71	0.58	-0.56
10.0	1.38	0.95	0.22	-1.04
30.0	1.40	0.98	0.13	-1.15

NOTE: $B' = B/\sqrt{N}$ and N = number of sensor map elements.

In Eq. (8), the ratio A is a function of the S/N ratio only, whereas the variable B is a function of both the S/N ratio and the number of sensor map elements. By defining a new variable $B' = B/\sqrt{N}$, as was done above, Eq. (8) can be rewritten in terms of variables that are either a function of only the S/N ratio (A and B') or the number of sensor map elements (N). The result is

$$P_c = 1/2 \left(\overset{+}{-}\right) \operatorname{erf} \left[\frac{(+)\ K - \sqrt{N}\ B'}{A}\right] \quad \begin{pmatrix}\max\\ \min\end{pmatrix} \text{ algorithm} \qquad (9)$$

With P_c expressed in this form, we can estimate the number of sensor elements required to achieve a given P_c level. Solving for N in terms of P_c and the other variables in the above equation yields

$$N = \left[\frac{(+)\ K - A\ \operatorname{erf}^{-1}\left[\left(\overset{-}{+}\right)\ (P_c - 1/2)\right]}{B}\right]^2 \quad \begin{pmatrix}\max\\ \min\end{pmatrix} \text{ algorithm} \qquad (10)$$

One must recognize that this approximation solves for the number of *independent* sensor elements in the scene, and that the number of pixels required may be several times the number of elements given by this equation to achieve a given P_c level if there is any spatial correlation in the scene.

Table 10 shows the value of N required to achieve a P_c close to unity ($P_c = 0.99$) using this expression, for both the MAD and Product algorithms. Also shown in this table is the value of N (taken from Ref. 1) for which the same P_c level is achieved. The approximation looks reasonably good except for the MAD algorithm cases with S/N ratio equal to 30. It should be remembered that this was the case for which the approximation for P_c also started to break down due to low values for A.

Thus the approximation technique can greatly simplify the P_c computation and provide, as well, a means for estimating a priori the number of sensor map elements required in the correlation process to achieve a given P_c level.

Table 10

APPROXIMATE NUMBER OF SENSOR MAP ELEMENTS REQUIRED TO ACHIEVE $P_c = 0.99$ ($Q = 10^5$)

Algorithm	S/N	$N_{Approximate}$	$N_{Simulated}$ (1)
Product	0.1	414	520
Product	1	84	100
Product	30	50	70
MAD	0.1	2523	3100
MAD	1	81	90
MAD	30	12	30

Returning to Eq. (9) with this approximation in hand, it is now possible to obtain a deeper insight into the importance of certain variables and their interaction in the correlation process. Since B'--the difference between the in- and out-of-register expected values of the correlation function normalized to $\sqrt{N}\,\sigma(\cdot)$--has the same sign as K, Eq. (9) can be written in a more general form (independent of whether the algorithm is maximizing or minimizing) as

$$P_c = 1/2 - \text{erf}\left(\frac{1}{A}\right)\left(|K| - \sqrt{N}\,|B'|\right) \tag{11}$$

Examination of this expression reveals that, to achieve a P_c greater than 1/2, it is necessary that

$$\sqrt{N}\,|B'| > |K| \tag{12}$$

and in fact that P_c can be maximized by minimizing the expression

$$\gamma = \left(\frac{1}{A}\right)\left(|K| - \sqrt{N}\,|B'|\right) \tag{13}$$

Table 11 shows some computed values of γ and P_c for three different algorithms. As seen in this table, minimum values of γ do correspond to maximum P_c values for a given S/N ratio. Several additional conclusions can also be tentatively drawn from these data and Eq. (13), as follows:

1. As the number of sensor map elements, N, increases, the value of γ decreases. This results in an increased value of P_c.
2. As the number of out-of-register positions, Q, increases, γ increases. This results in a decreased value of P_c.
3. Since A and B' are both dependent on the S/N ratio and the scene characteristics, no universal relationship between γ and these two variables is to be expected.
4. From the data given in Table 11 it appears, over the S/N ratio region examined, that the algorithm with the highest absolute value of B' also achieves the minimum value of γ, and hence

Table 11

COMPUTATION OF A, B', AND P_c FOR THE PRODUCT, MAD, AND PRODUCT-DIFFERENCE[a] ALGORITHMS

Algorithm	S/N	A	$1/\sigma(J)$	$[E\{\phi(J)\} - E\{\phi(J)\}]$	B'	γ ($\theta=10^3$, N=30)	P_c[b]
MAD	0.1	0.912	$1.52\sqrt{N}$	-0.076	-0.12	2.76	0.00
Product	0.1	1.042	$3.01\sqrt{N}$	0.10	0.301	1.47	0.07
Product-Difference	0.1	0.877	$2.78\sqrt{N}$	-0.10	-0.277	1.90	0.03
MAD	1.0	0.578	$0.96\sqrt{N}$	-0.584	-0.561	0.186	0.43
Product	1.0	1.22	$0.71\sqrt{N}$	1.0	0.707	-0.57	0.72
Product-Difference	1.0	0.50	$0.50\sqrt{N}$	-1.0	-0.50	0.88	0.19
MAD	30.0	0.128	$0.22\sqrt{N}$	-5.432	-1.15	-24.4	1.00
Product	30.0	1.402	$0.033\sqrt{N}$	30.0	0.983	-1.57	0.94
Product-Difference	30.0	0.1047	$0.0.9\sqrt{N}$	-30.0	0.574	+0.35	0.36

[a]The Product-Difference algorithm is defined as

$$\phi(J) = \frac{1}{N} \sum_{I=I}^{N} (X_{I+J} - Y_I)\, X_{I+J}$$

[b]P_c values are for a Gaussian scene with only an additive noise error.

the maximum value of P_c. Thus, the value of $|B'|$ can serve as a measure of the "goodness" of an algorithm, where, to repeat

$$|B'| = \left| \frac{E\{\phi(0)\} - E\{\phi(J)\}}{\sqrt{N}\,\sigma(J)} \right|$$

5. It does not appear from the data in Table 11 that further suboptimal expressions for maximizing P_c can be found by either maximizing the numerator or minimizing the denominator of $|B'|$.
6. It should also be noted from the data that, over the S/N ratio region investigated, minimizing the variable γ alone does not maximize P_c.

Reflection on the physical meaning of $|B'|$, as described at the top of page 20, makes it quite clear why its maximization is a good criterion, why suboptimization of its parts is not useful, and why it is far more efficient as an estimator than the ratio γ.

IV. ESTIMATION TECHNIQUES USING CORRELATION DATA

One concept frequently neglected in the correlation field is the possibility of using the correlation data themselves to estimate some of the important parameters of the process, such as the number of independent elements in the scene, N_I, the correlation length of the scene, ρ, the signal-to-noise ratio, S/N, and the probability of correct lock-on, P_c.

There are two methods by which one could obtain these estimates. First, one might develop an expression for the statistical properties of the correlation process, allowing the scene to be spatially correlated with any appropriate distribution. The problem here is that it is extremely difficult to develop these statistical relationships. However, if a general solution could be obtained, accurate estimates could be made directly. The other method would be to assume the scene to be uncorrelated and Gaussian distributed. The statistical expressions that result for any algorithm are generally quite simple; however, there is now an additional unknown quantity that needs to be estimated--the number of independent elements in the scene. Any spatial correlation in the scene will result in a lower number of independent elements than the number of pixels contained in the sensor map. This latter method is used to estimate the important parameters of the process, as described below.

ESTIMATING SCENE PARAMETERS FROM THE CORRELATION DATA

To illustrate the estimation process, let us first assume that in- and out-of-register values of the correlation function have been generated. The following quantities can be extracted from such data, where the subscript M indicates quantities derived from measured data:

$\phi_M(0)$ = minimum achieved value of the correlation function in-register

$\bar{\phi}_M(J)$ = average measured value of the correlation function out-of-register

$\sigma^2_M(J)$ = measured variance of the correlation function out-of-register

These quantities can then be used as estimates for the ensemble statistics $E\{\phi(0)\}$, $E\{\phi(J)\}$, and $\sigma^2(J)$, respectively. The quantity $\sigma^2_M(0)$ cannot be estimated from these data since, in general, there is only one observed extremum value. Quantities derived using these assumptions will carry a caret ($\hat{}$) over the symbol.

For the *MAD algorithm* (assuming an uncorrelated, Gaussian-distributed, zero-mean scene), the estimates of the statistical measures are related to the scene characteristics, σ^2_x, σ^2_n, and N_I, by the following equations:[1]

$$\bar{\phi}_M(0) = \sqrt{2/\pi}\ \hat{\sigma}_n \tag{14}$$

$$\bar{\phi}_M(J) = \sqrt{2/\pi}\ \left(2\hat{\sigma}^2_x + \hat{\sigma}^2_n\right)^{1/2} \tag{15}$$

$$\sigma^2_M(J) = (1 - 2/\pi)\frac{\left(2\hat{\sigma}^2_x + \hat{\sigma}^2_n\right)}{\hat{N}_I} \tag{16}$$

where σ^2_x = variance of the scene

σ^2_n = variance of the noise

N_I = number of independent elements in the sensor scene.

With three equations and three unknowns, one can thus compute estimates for the scene parameters from the measured data. Specifically, $\hat{\sigma}^2_n = (\pi/2)\ \phi^2_M(0)$ directly from Eq. (14), $2\hat{\sigma}^2_x = (\pi/2)\ [\bar{\phi}^2_M(J) - \phi^2_M(0)]$ from Eqs. (14) and (15), and the formula for $\hat{N}_I$ derived from Eqs. (14), (15), and (16) is

$$\hat{N}_I = \frac{(\pi/2 - 1)\ \bar{\phi}^2_M(J)}{\sigma^2_M(J)} \tag{17}$$

The entire procedure is diagrammed in Fig. 5. The two maps are first correlated. The extremum is taken as the in-register value of

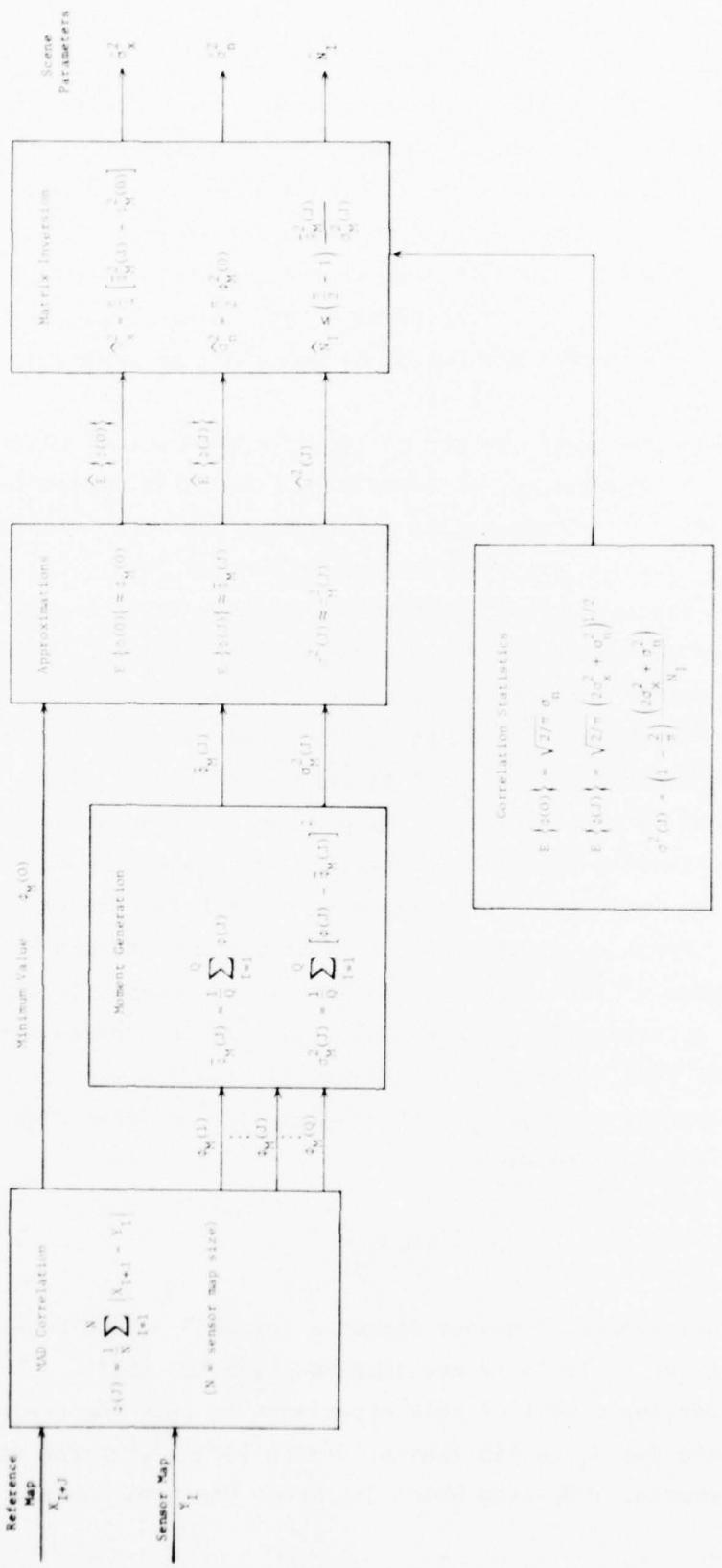


Fig. 5—Estimation of scene parameters from MAD correlation data

the correlation function and is extracted from the data; the first and second moments of the out-of-register values are then computed from the remaining data. Based on these moments, the extremum value of the correlation function, and the correlation statistics, the scene parameters are estimated. This estimation procedure assumes that the correlation statistics for the MAD algorithm are Gaussian. Obviously, most scenes are not truly Gaussian and do not generate purely Gaussian statistics. The question remains as to how close to Gaussian, in general, are typical scene statistics.

A control experiment was run to test the accuracy of the parameter estimates and to examine the validity of the Gaussian assumption. The experiment consisted of generating a reference map whose elements were independently Gaussian distributed, and extracting from this reference map a 10 × 10 element sensor map that was then correlated over the entire reference map using the MAD algorithm in the absence of noise. Based on the out-of-register correlation data, $\bar{\Phi}_M(J)$ and $\sigma_M^2(J)$ were computed and $\hat{N}_I$ estimated using Eq. (17). Table 12 shows the results of the estimation process using 25 different reference scenes, each of which contained 15 × 15 elements. Also shown in this table are the values of the third and fourth moments, μ_3 and μ_4, of the out-of-register correlation function. For a Gaussian distributed process, the third moment should equal zero and the fourth moment should be three times the square of the variance. As seen in the table, in this case the process is reasonably close to Gaussian. These moments were also calculated for real scene imagery and similar results were found. The correlation length, discussed further below, can be defined quite generally by the relationship

$$\hat{\rho} = \left(N/\hat{N}_I\right)^{1/2} \tag{18}$$

with N = actual number of sensor elements (pixels) in the scene. The values of $\hat{\rho}$ given in Table 12 are computed from Eq. (18).

The disturbing result of this experiment is that the average value of the estimate for N_I is 168 (which is much larger than the actual number of elements, 100, from which the scene was constructed) and that

Table 12

ESTIMATION OF $\hat{N}_I$ USING MAD ALGORITHM DATA

(M = 225, N = 100, Q = 35, random scene)

Run	Variance (out of register) $\sigma^2(J)$	Third Moment M_3	Fourth Moment $M_4/\sigma^4(J)$	Estimated No. of Independent Elements $\hat{N}_I$	Estimated Correlation Length $\hat{\rho}$
1	0.00424	-0.00003	2.325	148.0	0.822
2	0.00363	0.00002	2.933	200.9	0.705
3	0.00307	0.00001	2.253	238.1	0.648
4	0.00259	-0.00259	2.153	288.1	0.589
5	0.00439	0.00015	2.668	148.1	0.822
6	0.00557	-0.00015	2.536	137.5	0.852
7	0.00747	-0.00008	2.465	88.7	1.062
8	0.00324	-0.00014	3.826	259.0	0.621
9	0.00801	0.00017	2.052	94.9	1.027
10	0.00383	-0.00002	1.911	200.3	0.707
11	0.00753	0.00007	2.354	117.6	0.922
12	0.00398	0.00000	2.079	144.3	0.832
13	0.00405	-0.00002	2.679	163.1	0.783
14	0.00532	0.00001	3.380	134.7	0.862
15	0.00414	0.00002	2.621	159.6	0.792
16	0.00426	-0.00017	3.139	173.4	0.760
17	0.00528	-0.00000	2.637	134.5	0.862
18	0.00391	0.00006	2.574	164.2	0.780
19	0.00415	-0.00012	2.125	215.1	0.682
20	0.01144	0.00009	2.828	83.2	1.096
21	0.00475	0.00011	2.406	206.5	0.696
22	0.00347	0.00004	2.493	239.9	0.646
23	0.00656	-0.00011	2.286	114.6	0.934
24	0.00290	-0.00009	3.066	250.0	0.633
25	0.00680	0.00003	2.317	99.9	1.001
Mean	0.00498	-0.00007	2.564	168.2	0.805

there is a large variation in the range of the estimate (from 83 to 228). This experiment was repeated using different reference map sizes, with similar results. The hypothesis was put forth that there is nothing wrong with the estimation technique per se, but rather that values of the MAD algorithm computed at closely spaced displacements are, in fact, correlated. Such a correlation would cause the variance of the map-correlation function to be somewhat smaller than was

calculated by the theory, thus leading to overestimation of N_I through Eq. (17).

To show that there is indeed correlation between adjacent values of the correlation function when the MAD algorithm is used, a simple two-element sensor map was hypothesized and correlated over adjacent positions with a three-element reference map. The reference map elements, X_I, were considered to be independent and identically distributed with distribution $N(0, \sigma_X^2)$, and the sensor map elements, Y_I, had the distribution $N(0, \sigma_Y^2)$. The adjacent correlation functions were then defined to be

$$\phi(1) = |X_1 - Y_1| + |X_2 - Y_2| \tag{19}$$

$$\phi(2) = |X_2 - Y_1| + |X_3 - Y_2| \tag{20}$$

W. K. Chow* of The Rand Corporation has determined the covariance between these two functions to be

$$\text{Cov}\,[\phi(1), \phi(2)] = \frac{2}{\pi}\left(\sigma_X^2 + \sigma_Y^2\right)\left[2\left(\sqrt{1-\mu^2} + \mu \sin^{-1}\mu - 1\right) + \left(\sqrt{1-\nu^2} + \nu \sin^{-1}\nu - 1\right)\right] \tag{21}$$

where

$$\mu = \frac{\sigma_Y^2}{\sigma_X^2 + \sigma_Y^2} \quad \text{and} \quad \nu = \frac{\sigma_X^2}{\sigma_X^2 + \sigma_Y^2}$$

Since the correlation coefficient between these two functions, $\phi(1)$ and $\phi(2)$, is given by the covariance of the two functions divided by the product of the standard deviations of each (and, in practice, the

*Private communication.

covariance is almost never equal to zero), then, in general, there is correlation between these two adjacent values. Thus Eqs. (16) and (17) are *not* valid for the determination of scene parameters from MAD correlation data.

This correlation between adjacent computed values has a further important implication with regard to the predicted performance of the MAD algorithm. The reduction in the out-of-register variance not only leads to an overestimate of N_I (as mentioned), but also, and more importantly, results in an underestimate of P_c--hence the values calculated in the past for the MAD algorithm are all too conservative.

For the *Product algorithm* (again assuming an uncorrelated, Gaussian-distributed, zero-mean scene), the estimates of the statistical measures are related to the scene characteristics by the following equations:[1]

$$\phi_M(0) = \hat{\sigma}_x^2 \tag{22}$$

$$\bar{\phi}_M(J) = 0 \tag{23}$$

$$\sigma_M^2(J) = \left(\hat{\sigma}_x^2/\hat{N}_I\right)\left(\hat{\sigma}_x^2 + \hat{\sigma}_n^2\right) \tag{24}$$

Since Gaussian theory requires the average of the out-of-register values to vanish, the second of these equations is not usable in solving for the scene parameters; hence there are only two equations with three unknowns. However, in the noise-free case, i.e., in autocorrelations with $\sigma_n^2 = 0$, Eqs. (22) and (24) can be solved for $\hat{N}_I$ with the result

$$\hat{N}_I = \frac{\phi_M^2(0)}{\sigma_M^2(J)} \tag{25}$$

The complete process in this case is diagrammed in Fig. 6.

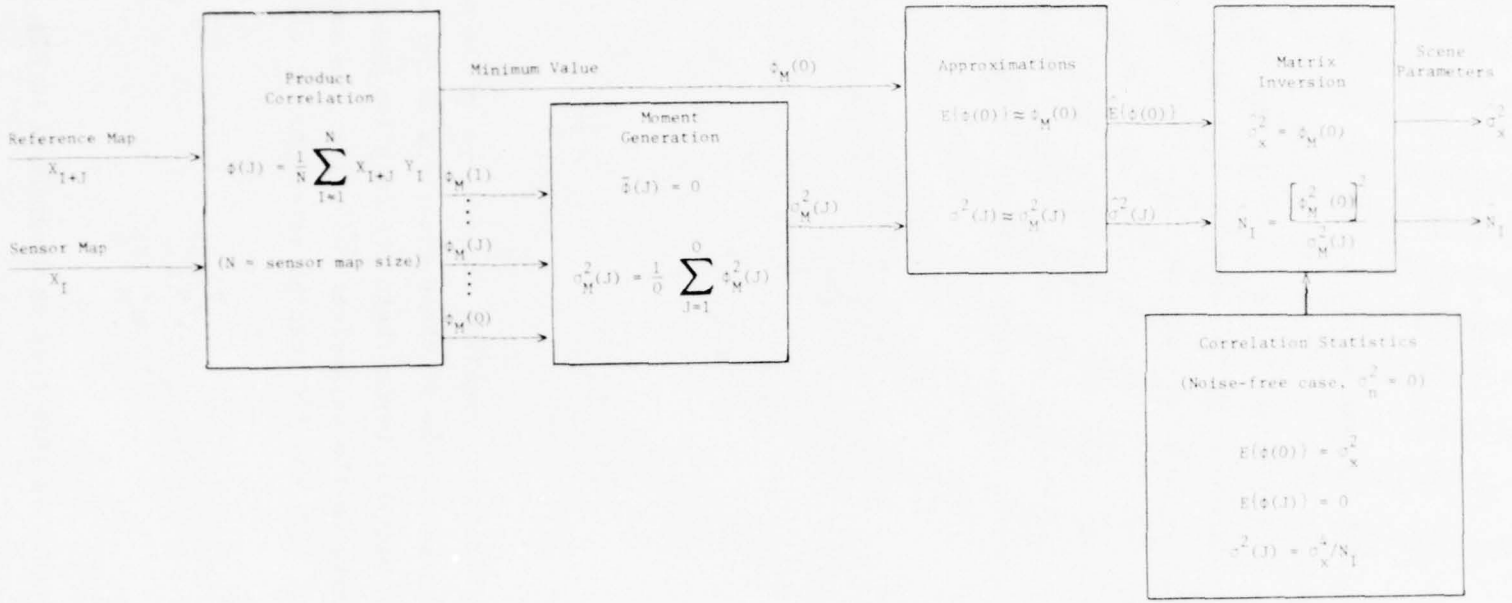


Fig. 6—Estimation of scene parameters from Product correlation data

(Noise-free case, $\sigma_n^2 = 0$)

Further analysis of a two-element sensor map shows that the covariance between adjacent values of the Product correlation function is zero, indicating (as expected) statistically independent values. The experimental results shown in Table 13 verify this analytical prediction. The average number of independent elements over all reference maps is close to 100, and the estimated correlation length, defined by Eq. (18), is close to unity. The assumption of Gaussian statistics was again tested using real-world imagery in Product correlation, and the result was again reasonably positive.

In the past, correlation length has normally been computed by determining the average distance over a given region of the scene at which the autocorrelation falls to a value of 37 percent of its maximum. This process is complicated when applied to a two-dimensional map because of the multiple directions in which the correlation function can be computed. The estimation method presented here is an operationally simpler technique for determining the effective correlation length of a scene. This technique was used to estimate the number of independent elements and the correlation length in four regions (agricultural, mountain, desert, and suburban) of the earth resources satellite data base described in Ref. 1. Each region was broken down into twenty-five 20 × 20 reference maps. The Product correlation algorithm was applied to each of these twenty-five subregions, using 10 × 10 sensor maps, and an estimate of the number of independent elements and the corresponding correlation length in each subregion was obtained. These estimates are shown in Tables 14 through 17 for the four regions. The average values of the correlation lengths in each of these regions are summarized below:

Region	$\hat{\rho}$
Agriculture	3.0
Mountain	2.7
Desert	2.0
Suburban	2.0
Random (from Table 13)	1.0

Table 13

ESTIMATION OF $\hat{N}_I$ AND $\hat{\rho}$ USING PRODUCT ALGORITHM
(M = 225, N = 100, Q = 35, random scene)

Subregion	Independent Elements $\hat{N}_I$	Correlation Length $\hat{\rho}$
1	104.04	0.98
2	91.30	1.05
3	91.88	1.04
4	74.77	1.16
5	136.41	0.86
6	99.41	1.00
7	102.08	0.99
8	94.28	1.03
9	91.22	1.05
10	90.23	1.05
11	127.51	0.89
12	103.46	0.98
13	116.23	0.93
14	114.32	0.94
15	93.91	1.03
16	92.33	1.04
17	91.87	1.04
18	143.66	0.83
19	80.80	1.11
20	86.26	1.08
21	85.78	1.08
22	97.48	1.01
23	125.43	0.90
24	111.15	0.95
25	87.45	1.07
Mean	101.25	1.004

ESTIMATING THE S/N RATIO FROM THE CORRELATION DATA

When the sensor map contains extraneous noise, as it always does, it is of considerable interest to estimate the S/N ratio directly from the correlation data. One can do so for the MAD algorithm by solving for estimates of σ_x^2 and σ_n^2 using Eqs. (14) and (15) and then taking the

Table 14

ESTIMATES FOR AGRICULTURAL SCENES

(Product, M = 400, N = 100)

Subregion	Independent Elements $\hat{N}_I$	Correlation Length $\hat{\rho}$
1	15.32	2.55
2	6.39	3.96
3	6.01	4.08
4	17.54	2.39
5	39.95	1.58
6	10.49	3.09
7	15.72	2.52
8	20.35	2.22
9	17.69	2.38
10	14.81	2.60
11	10.30	3.12
12	5.61	4.22
13	11.23	2.98
14	7.98	3.54
15	6.19	4.02
16	9.26	3.29
17	11.27	2.98
18	16.11	2.49
19	9.40	3.26
20	11.99	2.89
21	31.21	1.79
22	10.31	3.12
23	12.25	2.86
24	9.80	3.19
25	12.96	2.78
Mean	13.60	2.96

Table 15

ESTIMATES FOR MOUNTAIN SCENES

(Product, M = 400, N = 100)

Subregion	Independent Elements $\hat{N}_I$	Correlation Length $\hat{\rho}$
1	22.67	2.10
2	39.22	1.60
3	9.28	3.28
4	10.47	3.09
5	39.77	1.59
6	9.61	3.23
7	23.48	2.06
8	11.01	3.01
9	5.55	4.24
10	14.16	2.66
11	10.63	3.07
12	23.06	2.08
13	9.73	3.21
14	25.33	1.99
15	20.35	2.22
16	6.27	3.99
17	17.36	2.40
18	12.17	2.87
19	13.74	2.70
20	9.03	3.33
21	16.10	2.49
22	27.92	1.89
23	14.07	2.67
24	17.31	2.40
25	18.41	2.33
Mean	17.07	2.66

Table 16

ESTIMATES FOR DESERT SCENES

(Product, M = 400, N = 100)

Subregion	Independent Elements $\hat{N}_I$	Correlation Length $\hat{\rho}$
1	34.03	1.71
2	10.46	3.09
3	24.25	2.03
4	44.85	1.49
5	16.40	2.47
6	20.98	2.18
7	6.71	3.86
8	31.66	1.78
9	26.01	1.96
10	36.93	1.65
11	25.81	1.97
12	27.63	1.90
13	35.69	1.67
14	50.49	1.41
15	23.90	2.05
16	11.58	2.94
17	43.66	1.51
18	25.81	1.97
19	27.04	1.92
20	34.77	1.70
21	19.46	2.27
22	24.56	2.02
23	27.92	1.89
24	58.76	1.30
25	35.65	1.67
Mean	29.00	2.01

Table 17

ESTIMATES FOR SUBURBAN SCENES

(Product, M = 400, N = 100)

Subregion	Independent Elements $\hat{N}_I$	Correlation Length $\hat{\rho}$
1	31.68	1.78
2	21.32	2.17
3	22.21	2.12
4	46.27	1.47
5	30.28	1.82
6	25.65	1.97
7	17.89	2.36
8	14.78	2.60
9	31.24	1.79
10	25.98	1.96
11	25.37	1.99
12	27.47	1.91
13	35.76	1.67
14	26.27	1.95
15	24.84	2.01
16	27.98	1.89
17	24.18	2.03
18	12.55	2.82
19	26.36	1.95
20	12.82	2.79
21	54.44	1.36
22	19.38	2.27
23	24.60	2.02
24	24.06	2.04
25	33.55	1.73
Mean	26.68	2.02

ratio--avoiding the use of Eq. (16), since N_I is known to be incorrect. Table 18 shows the comparison between the estimated and actual S/N ratio for a 10 × 10 sensor map and a 20 × 20 reference map taken over various regions of the earth resources satellite map. The estimate is correct to within about 15 percent rms.

A more detailed description of this estimator and its variance, and analogous results for the Product algorithm, are contained in the appendix.

Table 18

COMPARISON OF ESTIMATED AND ACTUAL S/N RATIOS USING THE MAD ALGORITHM

(M = 400, N = 100, Q = 120)

Region	S/N (actual)	$\widehat{S/N}$ (estimated)
Agricultural	0.70	0.61
	0.97	1.04
	0.97	1.02
Mountain	1.07	1.12
	0.96	0.87
	0.92	1.06
Desert	1.13	0.95
	1.14	0.80
	0.92	0.76
Suburban	0.95	1.11
	1.12	0.98
	1.02	0.86
Random	0.99	0.88

ESTIMATING P_c FROM THE CORRELATION DATA

The most important quantity to be estimated from the correlation data is the degree of confidence one can have that the extremum gives the correct matchpoint, i.e., the P_c associated with the extremum. Using the approximation technique developed in Sec. III, P_c can be expressed, following Eq. (11), as

$$P_c = 1/2 - \text{erf}\left\{\left(\frac{\sigma(J)}{\sigma(0)}\right)\left[|K| - \left|\frac{E\{\phi(0)\} - E\{\phi(J)\}}{\sigma(J)}\right|\right]\right\} \qquad (26)$$

where K is given by Eq. (5) and A and B have values as defined following Eq. (2).

As before, the value of the extremum, $\phi_M(0)$, can serve as an estimate for $E\{\phi(0)\}$. Similarly, the mean value of all the out-of-register correlation values, $\bar{\phi}_M(J)$, and the variance of these values, $\sigma^2_M(J)$, measured from the data can serve as estimates for $E\{\phi(J)\}$ and $\sigma^2(J)$, respectively. In order to estimate P_c, it is still necessary to find values for $\sigma(0)$ and K in Eq. (26). The exact procedures to be used differ for the two principal algorithms.

For the *Product algorithm*, the crucial step is the generation of the estimate for N_I given by Eq. (25), which was obtained by performing an autocorrelation on the reference map. Actually, in this process one is using the values of N_I and, through Eq. (18), of the effective correlation length measured on the reference map as estimates for the corresponding quantities on the sensor map; accordingly, one should probably perform the autocorrelation on the sensor-map-sized portion of the reference centered on the target, or a somewhat larger sized portion, but not on the entire reference map.

With $\hat{N}_I$ thus in hand, and σ_x^2 and σ_n^2 determined from Eqs. (22) and (24), one can proceed. One first observes that, although $\sigma(0)$ cannot be obtained directly from the data (since there is ordinarily only one extremum), it is only the ratio $\sigma(J)/\sigma(0)$ that occurs in Eq. (26). Now $\sigma^2(J)$ is given by Eq. (24) and $\sigma^2(0)$ is known[1] to be $\left(2\sigma_x^2 + \sigma_n^2\right)\left(\sigma_x^2/N_I\right)$. Thus the desired ratio is simply

$$\frac{\sigma(J)}{\sigma(0)} = \left(\frac{\sigma_x^2 + \sigma_n^2}{2\sigma_x^2 + \sigma_n^2}\right)^{1/2} \qquad (27)$$

which involves only known quantities.

Secondly, from Eq. (5), K is a function only of Q, the number of nonmatching displacements for which the correlation function is evaluated. Q in turn is given in Ref. 1 (in the footnote on p. 7) as

$$Q = (m - n + 1)^2 - 1 \tag{28}$$

where $m^2 = M$ is the reference map size and $n^2 = N$ is the sensor map size. Since the sensor map size has been scaled down by the square of the correlation length to use the statistical relationships, it is also necessary to scale Q similarly. The expression for the number of independent displacements, Q_I, is therefore

$$Q_I = \left(m \sqrt{\frac{N_I}{N}} - \sqrt{N_I} + 1 \right)^2 - 1 \tag{29}$$

Thus Q and K can also be estimated, and then P_c using Eq. (26). This complete process is diagrammed in Fig. 7.

For the *MAD algorithm*, the ratio $\sigma(J)/\sigma(0)$ can be found by a similar process. $\sigma^2(J)$ is given in Eq. (16), and $\sigma^2(0)$ is known[1] to be $(1 - 2/\pi)\left(\sigma_n^2/N_I\right)$. Thus the desired ratio is

$$\frac{\sigma(J)}{\sigma(0)} = \left(\frac{2\sigma_x^2 + \sigma_n^2}{\sigma_n^2} \right)^{1/2} \tag{30}$$

However, K is still a function of Q_I, as given by Eq. (29). Since $m > n$, Q_I increases monotonically with N_I, and so does K by Eq. (5); and finally, P_c decreases with increasing K, as shown by Eq. (11) or Eq. (26). Thus it is seen that the previously mentioned fundamental difficulty with the MAD algorithm, caused by the partial correlation of adjacent values of this particular function, cannot be avoided. The values of N_I derived from the equations will always be somewhat too high, and the resulting predictions for P_c correspondingly pessimistic. The estimation procedure given here is nevertheless expected to be sufficiently accurate to be useful under many circumstances, and is

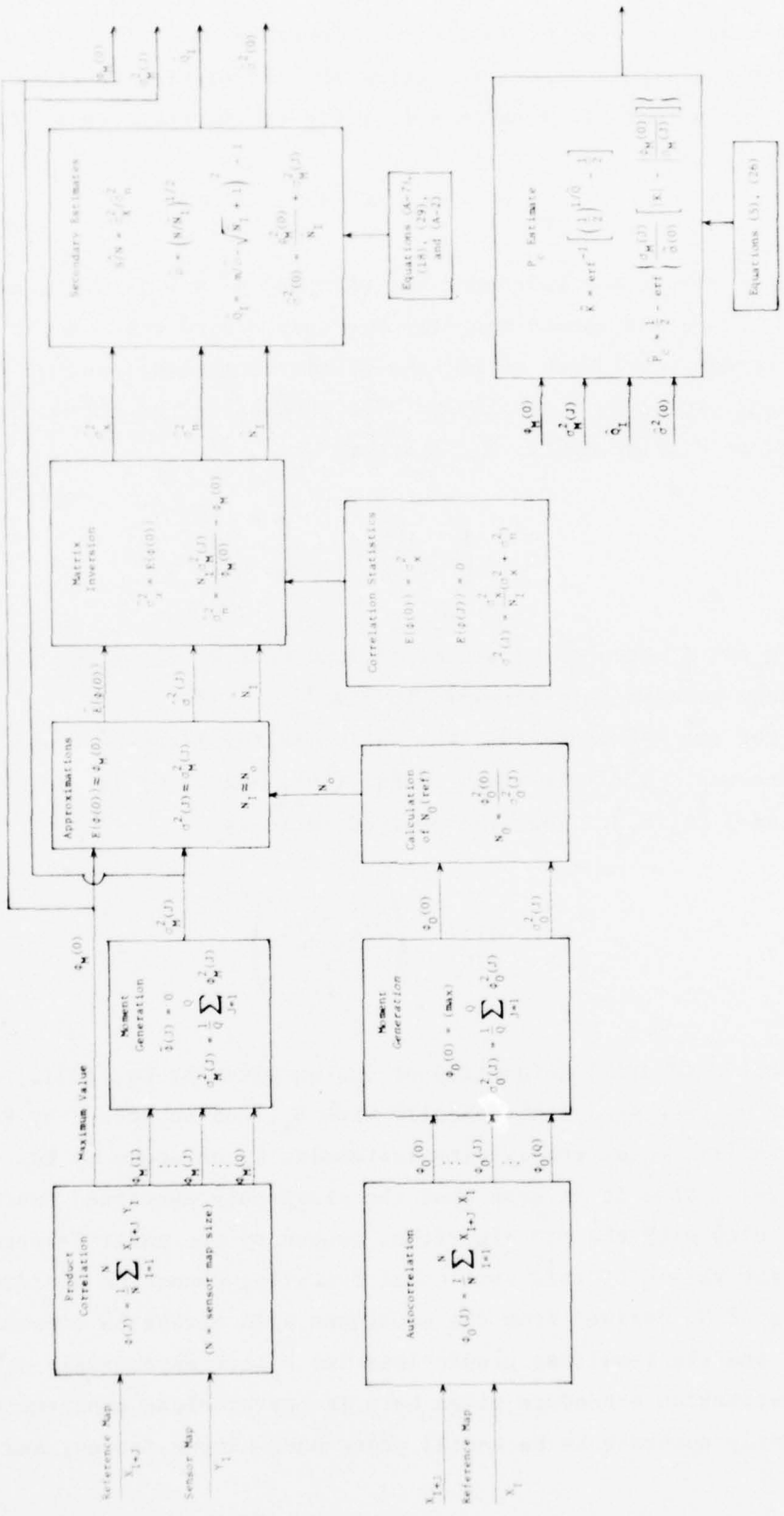


Fig. 7—P_c estimation process for Product algorithm

diagrammed in full in Fig. 8. The inherent difficulty is reflected in the use of an $\geq$ sign with $\hat{P}_c$ in the last box of the figure.

An experimental simulation of the estimation process has been run for an agricultural region of the earth resources satellite map. The MAD algorithm was employed to investigate the degree of overestimation of N_I and Q and the underestimation of P_c. The reference map size was 20 × 20 and the sensor map size was 10 × 10. Additive noise was superimposed on the sensor map such that the S/N ratio was approximately 1. Twenty-five correlations were run using the same scene and different noise samples. For this scene, N = 100 and Q = 120. Table 19 shows the results of this estimation experiment. For all 25 runs, the correlation algorithm (MAD) correctly matched the two maps (i.e., simulated $P_c = 1$). P_c was estimated using, first, the $\hat{Q}_I$ derived from estimating the number of independent sensor map elements, as given by Eq. (29), and second, the entire number of map matching positions (Q = 120), where no attempt was made to account for correlation in the scene. This latter approach obviously yields a very low (conservative) estimate of P_c, as expected. Although the former approach, $\hat{P}_c(\hat{Q}_I)$, is not quite right, since the estimates of N_I are still too high, the estimates for P_c are reasonably close to the simulated value of 1. The S/N ratio and the correlation length were also estimated in this experiment, and the values are included in Table 19. Note that this experiment (one scene with 25 different samples of noise added) is different from that in Table 14 (25 different scenes), and furthermore that the effective correlation length in the present case should be significantly less than that in Table 14 because the added noise is completely uncorrelated.

The experiment was extended by examining the P_c estimates when the wrong sensor map (a map not extracted from the reference map region) was correlated over the same agricultural region. The results are shown in Table 20 with two different noise samples added to make the S/N ratio of the map approximately 1.

In conclusion, the estimation technique appears capable of yielding an estimate for P_c from a single correlation function similar to that found by Monte Carlo methods over numerous correlation runs. In

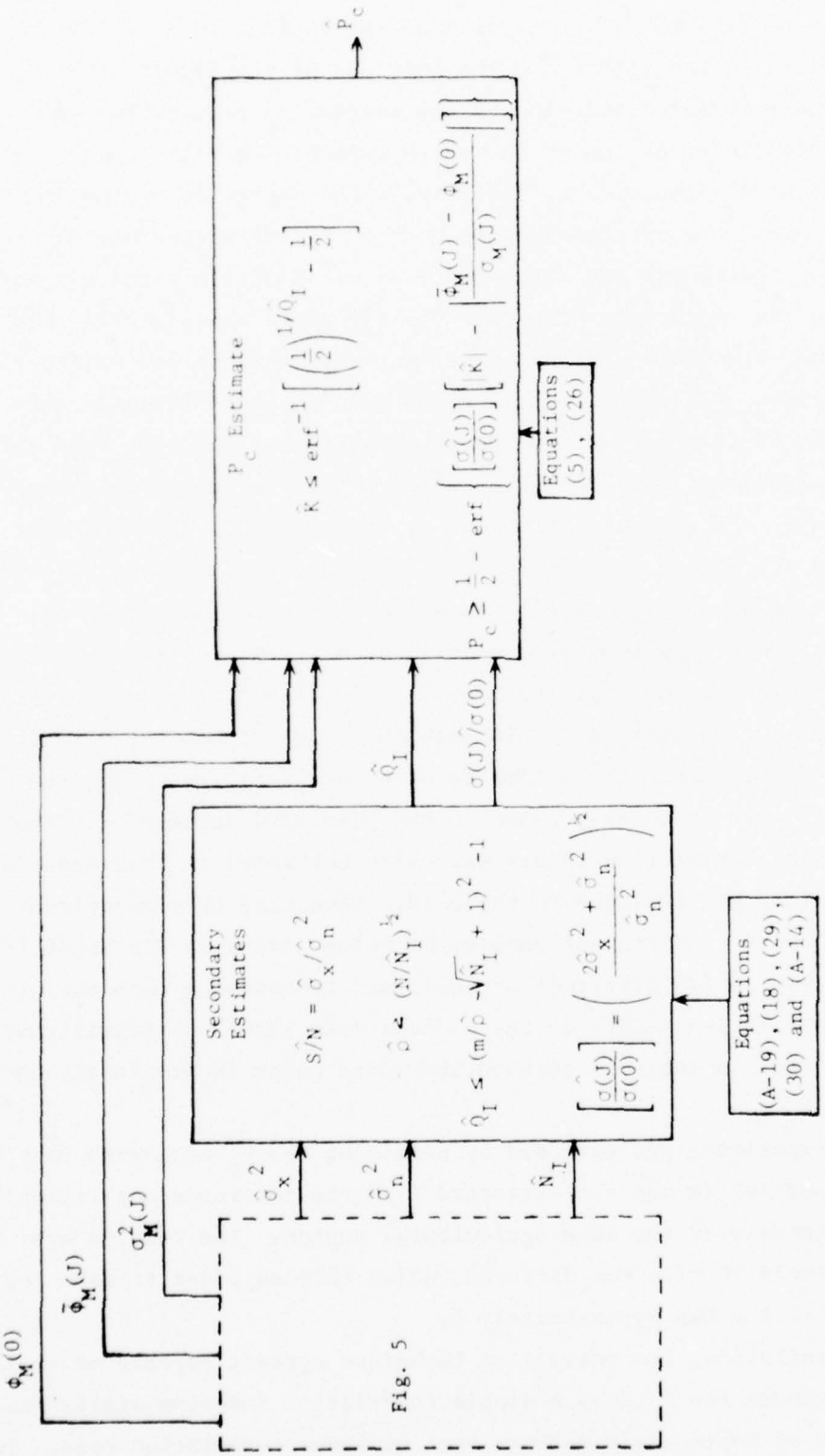


Fig. 8—P_c estimation process for MAD algorithm

Table 19

ESTIMATION OF P_c FOR AN AGRICULTURAL REGION OF THE EARTH RESOURCES MAP

(MAD, M = 400, N = 100, Q = 120, S/N = 1)

Run Number	N_I	Estimated Correlation Length	$\hat{S/N}$	S/N (actual)	Q_I	$\hat{P_c}(Q_I)$	$P_c(Q = 120)$
1	23.5	2.06	0.96	1.08	33.2	0.85	0.59
2	27.8	1.89	0.98	1.01	38.4	0.92	0.76
3	26.2	1.95	1.47	1.00	36.4	0.99	0.95
4	28.6	1.86	1.17	0.92	39.3	0.97	0.89
5	19.8	2.24	1.03	0.91	28.7	0.83	0.50
6	26.6	1.93	1.03	0.97	36.9	0.93	0.76
7	21.8	2.14	1.47	1.06	31.1	0.98	0.85
8	30.8	1.80	0.89	0.96	41.9	0.91	0.76
9	25.8	1.96	1.19	1.09	35.9	0.96	0.84
10	33.0	1.73	1.03	0.99	44.5	0.97	0.90
11	25.1	1.99	1.49	1.09	35.1	0.99	0.94
12	25.6	1.97	1.09	0.97	35.7	0.94	0.77
13	27.4	1.91	1.18	1.10	37.8	0.97	0.87
14	23.6	2.05	1.05	0.93	33.3	0.90	0.67
15	24.9	2.00	1.02	1.04	34.9	0.91	0.70
16	28.7	1.86	1.07	1.01	39.4	0.96	0.84
17	29.1	1.85	1.13	0.96	39.9	0.97	0.88
18	29.2	1.85	1.08	0.95	40.0	0.96	0.86
19	27.2	1.91	1.15	0.92	37.7	0.96	0.85
20	29.5	1.83	1.24	0.94	40.4	0.98	0.93
21	28.5	1.87	1.10	1.07	39.2	0.96	0.85
22	23.3	2.07	1.05	1.01	32.9	0.90	0.66
23	29.7	1.83	1.39	1.03	40.7	0.99	0.97
24	22.2	2.12	1.04	0.90	31.6	0.87	0.61
25	29.0	1.85	1.18	0.95	39.7	0.98	0.90
Mean	26.7	1.94	1.14	0.99	37.0	0.94	0.80

addition, comparison of Tables 19 and 20 indicates that the estimation procedure can easily distinguish between cases where the sensor map is or is not contained in the reference map.

ADDITIONAL PERSPECTIVE ON THE ESTIMATION PROCESS

Further discussion may be helpful at this point on the nature of the map-matching process and the significance of the estimation procedures described above.

Table 20

ESTIMATION RESULTS CORRELATING A WRONG SENSOR MAP OVER THE AGRICULTURAL REGION

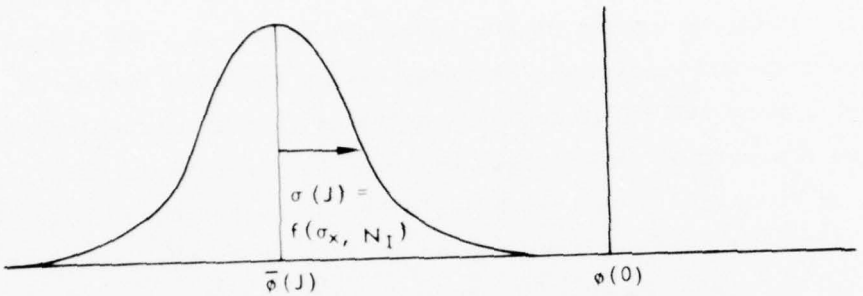
(N = 100, Q = 120)[a]

$\hat{N}_I$	$\hat{Q}$	$\hat{P}_c(\hat{Q})$	$\hat{P}_c(Q)$
49.1	63.1	0.225	0.149
48.5	52.6	0.235	0.155

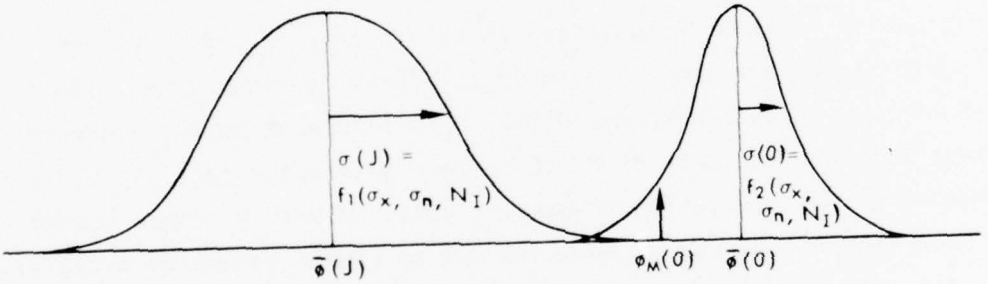
[a]Sensor scene does not come from reference scene.

Consider first the (auto)correlation of a single sensor map with an identical, noise-free, reference map. The distribution of values for this function is indicated schematically in Fig. 9A. For simplicity, a maximizing algorithm such as Product is assumed. It is further assumed that a digital calculation is performed and that the scene is spatially completely uncorrelated, so that the value at the match point is essentially a δ-function--otherwise there would be a small one-sided tail to the left of $\phi(0)$. In this case, the map-matching process is almost trivial. The spread in the out-of-register distribution function is determined by the statistics of the scene itself, a function of σ_x and N_I as given by Eq. (22) with $\sigma_n = 0$, and in principle this can be computed beforehand from the reference map.

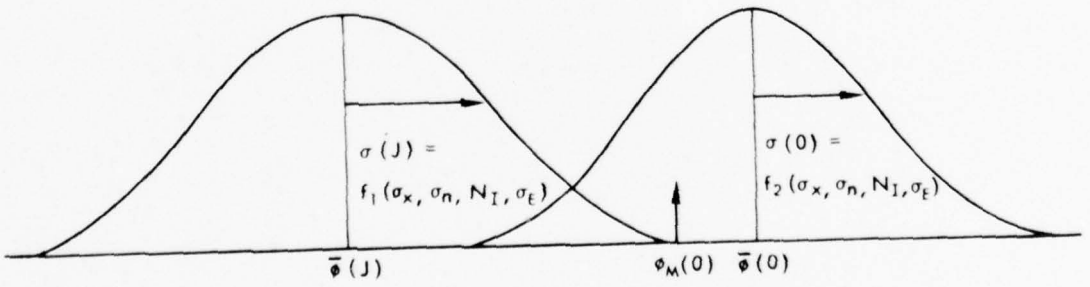
Consider next the introduction of noise--intensity fluctuations or prediction errors, geometrical distortions, real changes in the scene, all the causes of differences between the two maps--but still assuming a single sensor map properly centered over the target (i.e., with perfect midcourse navigation). The distribution of values of the correlation function is now represented in Fig. 9B. The possible crossover of the tails of the two curves demonstrates the possibility of a false match or gross error, and hence the need for calculating P_c, the probability of a correct match. This quantity can be calculated fairly easily if the distributions are assumed to be Gaussian. Since σ_x and N_I are known ahead of time, in this case the observed width of $\sigma(J)$



A. Autocorrelation with a single map (no noise)



B. Correlation with a single map with noise added



C. Correlation with an ensemble of possible sensor maps

Fig. 9 — Distribution of values of the correlation function under various conditions

could be used to estimate σ_n, from which P_c can be found. This is not, however, the real operational situation.

Consider finally the case in which there is a midcourse navigation error and the sensor map may be any one of a large number whose center is displaced from the center of the reference map (i.e., the target) by an unknown amount--an amount that may easily be large enough to cover a portion of the reference map having different statistical properties from the portion immediately adjacent to the target. In this case, one is concerned with the ensemble of possible sensor maps. The resulting distributions are represented in Fig. 9C by the artificial insertion of an extra "ensemble" term, σ_E, in the variances. It is these ensemble distributions that were analyzed by Johnson,[4] and extended by the Rand group,[2] in terms of parameters assumed to be known. The estimation procedures described in this report provide a method for drawing from the observed data an inference concerning the distribution of the possible extrema, $\sigma(0)$. This inference in turn permits the best possible estimate of P_c to be made before the fact, i.e., on the basis of the correlation statistics obtained with a single sensor map, even though $\bar{\phi}(0)$ cannot be known and is simply estimated by $\phi_M(0)$.

This estimation procedure is believed to represent a new contribution to the theory of image correlation.

V. INITIAL WORK IN FEATURE SELECTION

INTRODUCTION

Previous work has focused on area correlation per se, and Ref. 1 describes in considerable detail approaches to correlation from both theoretical and experimental points of view. It seems wise at this juncture to take a broader point of view and recognize that both the incorporation of feature selection methods from the field of pattern recognition and the use of more sophisticated algorithms can improve the map-matching process. Although the map-matching problem addressed here is not formulated precisely according to the standard pattern recognition paradigm of assignment to one of several classes, many concepts nevertheless carry over and, in particular, the techniques of feature extraction are relevant. First, however, we look at the noise characterization in our experiments, because this is important in assessing the relevance of the work for the real world.

NOISE IN THE MAPS

Most of the experimental work described in Ref. 1 assumed Gaussian noise. Although useful results can, and did, follow from this assumption, it is appropriate to look at noise that is more representative of that found in the real world. This is accomplished when a reference map is derived from reconnaissance imagery taken in one region of the spectrum (e.g., aerial photography) and the sensor operates in a different region (e.g., infrared or radar). In such cases, the major source of "noise" resides in erroneous predictions of the intensities placed in the reference map rather than in geometrical distortions. There were available several high-resolution pictures (7.8 meters per pixel) of downtown Los Angeles taken at various wavelengths in the visible and near-infrared portions of the spectrum, and it was clear that one of these pictures could serve as a "noisy" reference map and the other as a "corresponding" sensor map. Experiments with the two pictures in the yellow and near-infrared wavelengths yielded widely varying S/N ratios, with a mean of 1.2. The actual Pearson correlation

coefficient (related to the normalized product correlation) between the two pictures was only .59, despite the fact that the two pictures are registered identically and appear quite similar to the eye. This last observation emphasizes the fact that judicious feature selection is critical, as is discussed next.

FOUR EXPERIMENTS

It has long been recognized that careful feature selection is a key to effective pattern recognition. In conventional pattern recognition, feature selection (1) reduces the dimensionality of the space and hence the quantity of computing required and (2) improves the accuracy of classification. Although (1) has not been as important in this work, it is still relevant; item (2) is of obvious significance. Several different kinds of very preliminary feature-selection experiments have been conducted, as described below.

Edge Detection

One of the most obvious kinds of features to consider in map matching is that of edges. As described in Ref. 1, some simple edge detections were tried, using various gradient and Laplacian operators. The results were marginal, probably due to a combination of the noisiness of the pictures and the lack of sophistication in the operators themselves. More recently, the Hueckel operator[5] was applied to some of our pictures. Each picture was covered with a set of non-overlapping discs of constant diameter. In each disc, the Hueckel operator finds the best "ideal edge," that is, the straight line in the disc that best separates the disc into two parts of differing pixel value. Such an edge can have any orientation and any location within the disc, and in fact these values are outputs from the operator. A parameter of the Hueckel operator is the goodness of fit, that is, the degree to which an edge approximates an ideal edge. Experiments were conducted with various sizes of the Hueckel operator and values of the fit parameter. Two outputs are shown in Figs. 10 and 11, where the disc size was five pixels in diameter, i.e., a 5×5 square with the corners removed. The field boundaries in the agricultural region show clearly, whereas the

Fig. 10 — Hueckel operator applied to an agricultural scene

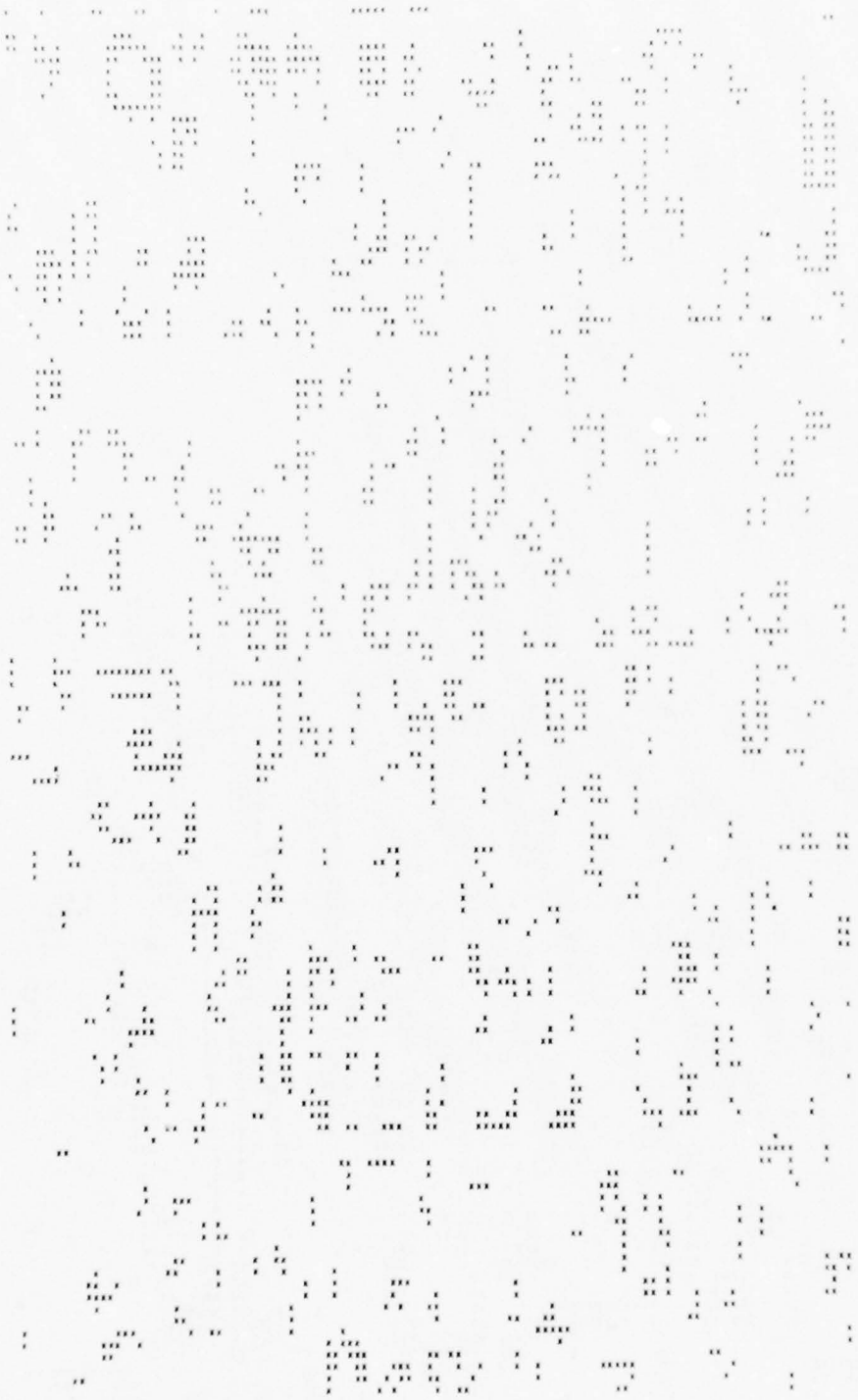


Fig. 11 — Hueckel operator applied to a suburban scene

features (including roads) of the suburban scene are vague for the most part.

We experimented further with Hueckel operators of different sizes, although the 5 × 5 square seemed best for our purposes. We also tried various values of the acceptance threshold. It is clear that there should be different values of this parameter and different sizes of the disc for different types of scenes, and that these parameters can be deduced experimentally. However, it would be valuable to have dynamic and automatic adjustment of these parameters, and experimentation with this concept seems in order.

A disappointing result of applying the Hueckel operator to the yellow and infrared suburban scenes was that the correlation between the scenes was much less (.29 compared with .59) after such application. This implies that correlation algorithms such as MAD and Product would not work as well after using the Hueckel operator. However, adjusting the acceptance threshold and operator size might well improve these results.

Other edge and shape detectors will be examined. In suburban scenes, for example, where there is a good estimate of road width (and even orientation in some cases), specialized detectors that use this information can be applied. This is part of a larger concept in which a whole arsenal of techniques is used to extract features from pictures, and some way is needed of reasonably selecting the best technique for application to any particular picture or class of pictures. The more automated this process can become, the better, although at some point the problem becomes close to one of artificial intelligence. This point is raised again later.

A Gradient Approach that Ensures the Presence of Features

A particular difficulty in depending on features, as above, is that a given scene may not contain any. Sensor maps are usually rather small, and it is quite possible that in relatively featureless areas the operators will yield nothing. As mentioned before, one approach to this problem is to modify the operator threshold or size. Another approach, which has been briefly tried, is to ensure features by

associating with each point some function of it and its neighboring points, such as the simple gradient or Laplacian. Specifically, the scheme associates with each point of a picture a vector of eight values, obtained as follows. Divide the 9×9 square of which the given point is the center into four 5×5 quadrants (the edges overlap). Then compute a smoothed gradient for each of these quadrants, with the origin at the center of each. Since each gradient has an x-value and a y-value, there is a vector of eight values for each point in the sensor scene. Such a vector obtained from a sensor scene can be compared with the corresponding stored (or computed) vectors for each point of a reference scene. The match point is determined to be that for which the associated reference vector is closest to the sensor vector, where "closest" can be any of a number of measures, including Euclidean distance, sum of the absolute deviations, minimax, or other. Preliminary tests actually gave poorer results in a simulation (19 correct matches out of 100, compared with 69 out of 100 for MAD), but clearly there is an opportunity to try different gradient sizes and different decision rules about vector comparison. Second-derivative information and other measures such as the goodness of fit of a Hueckel operator might also be used.

Features Based on Entropy

The amount of information conveyed by parts of a scene seems to be a plausible quantity on which to base feature selection, and indeed a measure based upon entropy considerations (directly related to information) is sometimes used in ordinary pattern recognition. A simple entropy calculation that was tried consisted of replacing the center point of a 5×5 square by the average information conveyed by the points in this square. This is done by counting the frequency of pixel values among the 25 points. We regard these frequencies as probabilities, i.e., if a particular pixel value i occurs K times, its empirical probability of occurrence is $p_i = K/25$. Then the total information is

$$-\Sigma p_i \log p_i$$

This calculation was made on the original suburban scene, after which correlation was simulated. The results were again worse than those for the original MAD algorithm (21 versus 69 cases out of 100) for S/N = 1 with a 5 × 5 sensor scene and a 15 × 15 reference scene. More judicious aggregation into classes or more elaborate information valuation (such as the concept of mutual information adapted for this situation) might improve these results.

It is worth noticing that, unlike most of the other features discussed here, entropy is a function of the distribution of pixel values rather than a direct function of the values themselves. As such, it represents another class of features and gives further insight into the various kinds of features available.

Features Based on the Estimated Value of P_c

Yet another approach to feature selection has been tried. In this new technique, for a given number of elements (feature size) to be chosen from the sensor map, all possible submaps contained in the sensor map are created. Each submap is then compared with the reference map using the correlation process; the match position and correlation statistics are simultaneously determined. The Gaussian theory then relates the statistics of the correlation process (expected value and variance of the in- and out-of-register values of the correlation function) to the scene parameters (variance of the scene intensity, variance of the noise, and number of independent elements in the scene). By using these relationships, it is possible to derive all the information necessary for estimating P_c, as described in Sec. IV. Having developed a technique for estimating P_c for each submap, it is now possible, by ranking the submaps relative to the P_c measure, to determine the features (sets of points contained in the submap) that are most significant in the scene.

One brief experiment has been performed that illustrates this concept and offers some encouraging results. Taking a very small simulated reference and sensor scene, it was attempted to determine the relative importance of each of the sensor points in making a potential match. This was done in two ways: One, through a direct simulation with noise

added, resulted in what is here called an "observed P_c"; the other, through the closed-form approximation described above, yielded an "estimated P_c."

The reference scene chosen was the following 7 × 7 "manufactured" scene containing only 0's, 1's, and 2's:

1	0	0	2	2	1	0
0	1	0	2	1	0	1
1	0	0	2	2	1	0
0	1	0	1	2	0	0
0	0	1	2	1	0	1
1	2	2	0	0	0	0
0	2	1	0	1	1	0

Fig. 12 — "Manufactured" reference scene

It has an ill-defined, approximately vertical streak that might be termed a "road." The center 3 × 3 portion of this scene was extracted and approximately 500 different samples of Gaussian noise were added to each of the cases formed by deleting exactly one point. The noise was multiplied by the appropriate factor with the MAD algorithm to give a S/N ratio of 1 for one set of cases and a S/N ratio of 3 for another set. The reduced 3 × 3 square (with noise added) was then superimposed in all possible positions. The "observed P_c" in Table 21 is the number of times the MAD algorithm yielded the correct match (with S/N = 1) divided by the total number of cases. The "estimated P_c" is the value calculated following the procedure described in Sec. IV.

The results show that single-point deletion in general has little effect, with the possibly significant exceptions that results were better when one particular point was deleted (8 or 1, depending on which method was used), and that they were definitely worse when another was deleted (9 or 2). The latter result indicates that point 9 (or 2) is the most "significant" in contributing to the correlation value.

The experiment was repeated deleting all possible *pairs* of points. The results were similar but more pronounced than for the single-point situation; they are shown in Table 22. A particular pair of points

Table 21

EFFECTS OF SINGLE-POINT DELETION

Point Deleted	Observed P_c	Estimated P_c
8	0.617	0.670
1	0.585	0.740
4	0.577	0.666
7	0.569	0.658
6	0.565	0.694
3	0.550	0.688
2	0.538	0.611
5	0.529	0.649
9	0.468	0.635
No deletion	0.591	

NOTE: Sensor scene points are numbered as follows:

1 2 3
4 5 6
7 8 9

(5, 9) was found to be the most important using the simulation method, and (2, 9) was the most important in the estimation approach. In both cases, the pair of points chosen were the two highest in importance in the corresponding single-point experiment.

A comparison of the two sets of results provides a crude measure of the "goodness" of the approximation approach. The estimated P_c values are all higher since no noise was added, but this is less important than the rank ordering. It is seen, for example, that the two least important pairs are the same in both approaches (though interchanged in order); the ordering at the upper end is not quite as consistent, although the same general trend is evident. The complete rank difference correlation coefficients are given in Table 23.

This experiment has demonstrated in a crude way the concept of selecting features on the basis of their P_c values. In principle, of course, we would continue the process of eliminating triplets, quadruplets, and so forth, of points from a real scene in the search for "features" that make significant contributions to P_c. Presumably, after a simple start as shown above, we might avoid the theoretical

Table 22

EFFECTS OF TWO-POINT DELETION

Point Pair Deleted	Observed P_c	Estimated P_c	Point Pair Deleted	Observed P_c	Estimated P_c
1,8	0.552	0.741	4,6	0.458	0.661
1,4	0.540	0.810	4,5	0.450	0.626
3,8	0.532	0.692	1,9	0.448	0.658
4,8	0.514	0.649	8,9	0.448	0.624
4,7	0.512	0.647	5,7	0.444	0.627
1,7	0.512	0.713	2,6	0.442	0.625
3,4	0.506	0.666	3,6	0.440	0.692
3,5	0.504	0.654	2,5	0.429	0.588
7,8	0.504	0.645	1,2	0.427	0.644
1,3	0.502	0.715	2,4	0.423	0.606
5,6	0.490	0.660	7,9	0.419	0.615
6,8	0.490	0.691	2,8	0.403	0.585
1,5	0.484	0.709	2,7	0.399	0.591
5,8	0.478	0.637	4,9	0.385	0.607
3,7	0.470	0.661	2,9	0.385	0.581
1,6	0.470	0.725	3,9	0.373	0.641
6,7	0.466	0.666	2,3	0.357	0.612
6,9	0.464	0.652	5,9	0.280	0.603

Table 23

RANK CORRELATIONS IN POINT DELETION EXPERIMENTS

Quantities to be Rank-Correlated	Number of Points Deleted	
	1	2
Estimated P_c versus observed P_c for S/N = 1	0.65	0.76
Estimated P_c versus observed P_c for S/N = 3	0.47	0.54
Observed P_c's for S/N = 1 versus S/N = 3	0.65	0.61

requirement to examine all possible combinations of points, concentrating instead only on those combinations that contain, for example, the best third of the smaller sets. This is but the germ of an idea; however, the modest success achieved at this point indicates the desirability of more extensive similar experiments in the future.

CONCLUSIONS

Two conclusions can be drawn from the three unsuccessful experiments. The first is that too simple and naive an application of conventional feature-selecting algorithms is not often adequate when dealing with images of real terrain. Those successes that have been demonstrated by others should be recognized as the result of quite sophisticated and extensive efforts. The second and more important conclusion is that all feature-selection algorithms, whatever their degree of success in achieving matches, should be conscientiously compared with the more conventional correlation alternatives for (a) accuracy, (b) gross error rate, and (c) cost in terms of computing effort.

The fourth approach, based on the use of P_c, is considered promising and further experimentation is recommended.

VI. IDEAS FOR FUTURE WORK

Three of the approaches described in Sec. V failed in their first implementation. They will be pursued, by adjustment of parameters or more sophisticated refinements, until either they are made to work or their failure is better understood. In addition to the feature selection based on P_c, there are at least two other techniques to be considered.

IMPROVING CORRELATION SYSTEMS

Figure 13 shows the layout of a typical correlation processing scheme in which sensor data are fed through a preprocessing operation. This operation may have many functions, including "grey level coding" of the data, feature extraction via pattern recognition techniques, and scene enhancement via either global (e.g., histogram equalization, Fourier transformation, etc.) or local operators (e.g., gradients, Laplacians, etc.). Once these sensor data are preprocessed, they are fed into a processor where, by means of a correlation algorithm, the match position and the location of the sensor map relative to the reference map are determined.

One of the major problems associated with this formulation of the correlation process is that if the match position (selected on the basis of some metric exhibiting an extremum) does not fall in the region surrounding the true match position, the correlator will fail. It would, therefore, be very desirable, in addition to determining the "match position," to also determine a confidence measure on which to base a determination that this is the true match position. The probability of correct lock-on, P_c, provides such a measure. However, its computation until now was a complicated calculation that was not obtainable in a closed-form expression. Section III has provided a "good approximation" method for obtaining P_c in a closed-form expression. This development has led to a reconfiguration of the correlation processor.

Figure 14 shows the proposed correlation scheme. It differs from

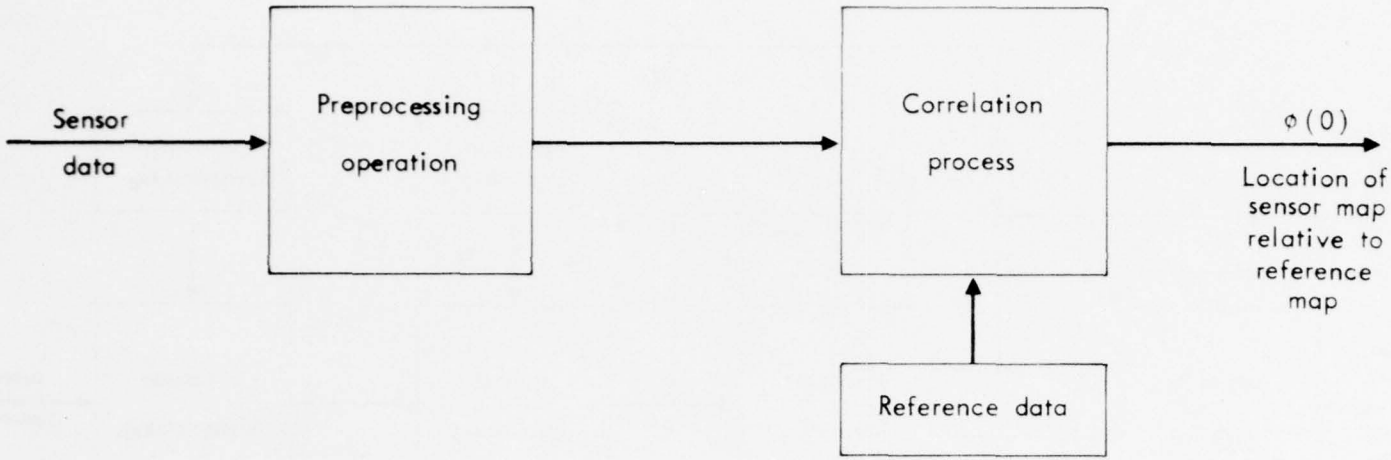


Fig. 13 — Present day correlation processing schemes (acquisition phase)

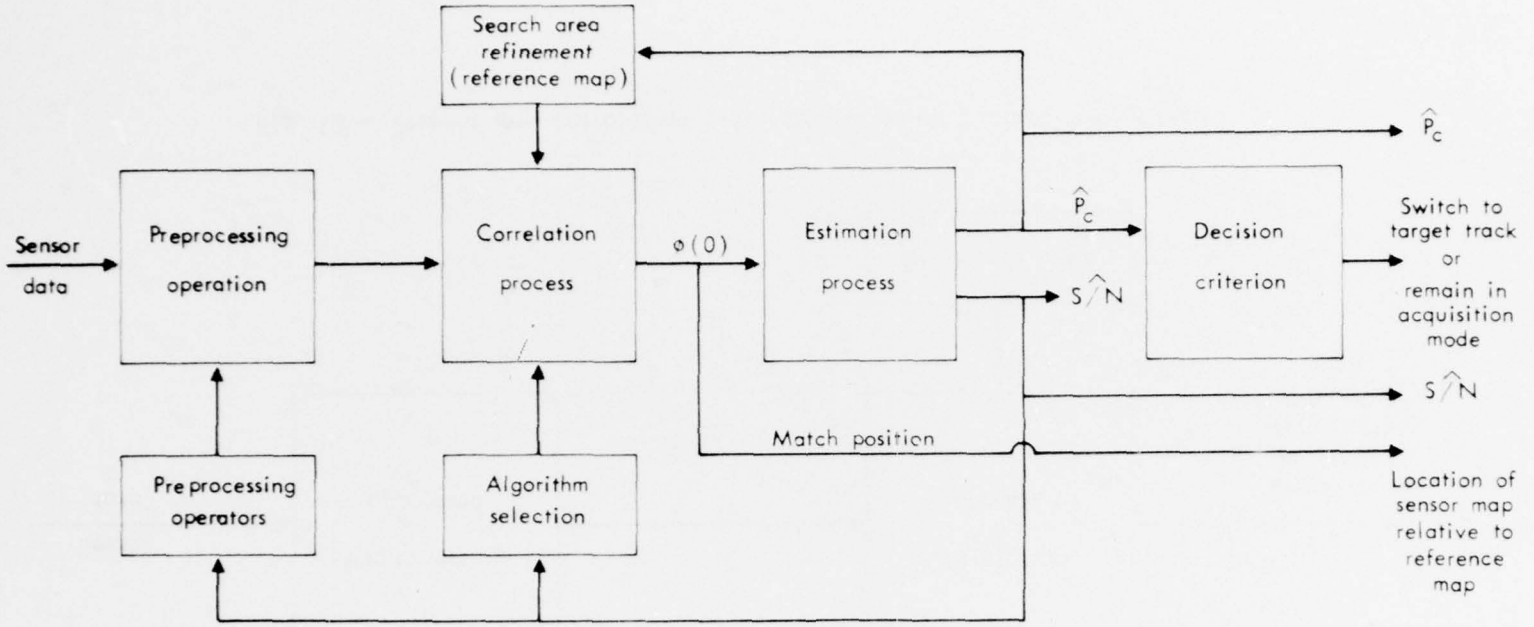


Fig. 14—Proposed next-generation processing schemes (acquisition phase)

the present-day schemes primarily in that an estimation process and decision criteria are included in the overall operation. The function of the estimation process is to determine from the correlation data what the scene statistics were (variance of the scene intensity and variance of the noise) and, based on these statistics, what are the best estimates of the S/N ratio and the probability of correct lock-on. Two assumptions are made in this estimation process. First, it is assumed that the scene statistics are Gaussian. The effects on this estimation process of non-Gaussian scene statistics have been tested but need more simulation using real-scene imagery data. Second, it is assumed that error sources other than additive noise (geometric, intensity scaling, etc.) are controlled to such an extent (by placing tolerance limits on design parameters) that they do not significantly alter the estimates obtained for $\hat{P}_c$ or $\hat{S/N}$. This assumption will also be tested using a simulation process.

The estimate of the S/N ratio can be valuable in selecting the proper algorithm for the next correlation update and possibly enhancing the performance of the preprocessing operation. The estimate of P_c is important from two standpoints. Most importantly, it can be useful in deciding whether the match position obtained via the correlation process is the true match position or not, and consequently whether the correlator should remain in the acquisition mode or should switch over to track mode. Second, if P_c is quite high, then, in addition to deciding to switch to a track mode, it would be possible to reduce the search area (which would free the computer to process out other error sources in the track mode); or, if P_c is low, then the search area might either remain fixed or possibly be expanded.

The success of this formulation of the correlation process depends on two conditions, namely: (1) that the two aforementioned assumptions (Gaussian statistics and limited geometrical errors) do not severely degrade the estimate of $\hat{P}_c$ and $\hat{S/N}$ and (2) that the variance of these estimates is small enough to render them useful.

Methods for estimating $\hat{P}_c$ and $\hat{S/N}$ from the data have been formulated. Initial analysis on the S/N estimate has shown it to have a relatively "tolerable" variance. Analysis on the variance of $\hat{P}_c$ and

testing these estimation techniques using simulation techniques remain to be done.

ARTIFICIAL INTELLIGENCE

Artificial intelligence was mentioned in Sec. V, and is discussed briefly here even though the precise level of application is not clear at this time.

A human performs map matching by bringing, if necessary, a vast amount of prior knowledge to bear on the problem; additionally, many specialized receptors in the eye and subsequent visual interpretation in the brain are powerful front-end processors. To simulate these processes directly is far beyond the state of the art; in fact, if this could be done, a significant part of the artificial intelligence problem would be solved and a new computer revolution would be at hand.

Nevertheless, techniques from the emerging computer science discipline of artificial intelligence can be applied to map matching. We have already mentioned edge detection. One subject within artificial intelligence describes the ways in which edges, lines, corners, and other primitives are put together to form recognizable objects. The analyst then matches these descriptions rather than the objects themselves.

Of ultimate importance in artificial intelligence is a "world model" that ties together a large amount of information about how real-world scenes of interest are constructed and how their subparts are related. It is a higher-order synthesis than that described above. For example, statements like "a large river always ends at an ocean" and "freeways never dead-end" are common-sense facts that people "know" but an artificial intelligence computer has to be told. Determining what statements are important, how to put them into an appropriate information structure, how to relate the statements, and how to access and manipulate the information are all major considerations in artificial intelligence.

Artificial intelligence programs are generally heuristic in nature, meaning that they try a number of plausible chains of reasoning in attempting to solve problems, and may back up and re-try several times

in the process. In map matching, for example, such programs could form hypotheses about the correct match, and then reject the hypotheses or test them further based on information in the stored world model.

Such artificial intelligence programs are large, complex, time-consuming to run, and expensive. These attributes may make them impractical for use in the real world of target acquisition. However, the decreasing cost and size of computers and the increasing understanding of artificial intelligence may make such solutions feasible within, say, the next decade or less. Even if other cost-effective methods are found in the meantime, an appreciation of the philosophy and techniques of artificial intelligence may lend some insight and direction into more conventional approaches.

Appendix
ESTIMATION OF THE S/N RATIO

This appendix describes a method for estimating the S/N ratio from the correlation data, using either the MAD or the Product algorithm, and determines the standard deviation of these estimates. It is noted that the estimates derived using the Product algorithm assume no spatial correlation in the scene, i.e., all map pixels are assumed to be independent (or N must be interpreted as the number of independent elements). The estimates using the MAD algorithms are also correct (unbiased), despite the difficulty with N for this algorithm that is discussed in Sec. IV, since the two equations that involve N [Eqs. (A-14) and (A-16)] do not have to be used.

ESTIMATING THE S/N RATIO USING PRODUCT ALGORITHM DATA

Statistically, for the Product algorithm, the following relationships hold true:

$$E\{\phi(0)\} = \sigma_x^2 \tag{A-1}$$

$$\sigma^2(0) = \frac{\sigma_x^2}{N}\left(2\sigma_x^2 + \sigma_n^2\right) \tag{A-2}$$

$$E\{\phi(J)\} = 0 \tag{A-3}$$

$$\sigma^2(J) = \frac{\sigma_x^2}{N}\left(\sigma_x^2 + \sigma_n^2\right) \tag{A-4}$$

From the correlation data, estimates for three of these quantities can be obtained as follows:

$E\{\phi(0)\}$ The value of the correlation function at the match position, $\phi_M(0)$, can serve as an estimate of this quantity. Since there is (generally) only one match point, there is no way of estimating $\sigma^2(0)$.

$E\{\phi(J)\}$ The mean of the measured out-of-register values of the correlation function can serve as the estimate of this quantity, i.e.,

$$\hat{E}\{\phi(J)\} = \frac{1}{Q}\sum_{J=1}^{Q} \phi_M(J) \triangleq \bar{\phi}_M(J)^* \qquad (A\text{-}5)$$

$\sigma^2(J)$ This quantity can be estimated by

$$\hat{\sigma}^2(J) = \frac{1}{Q}\sum_{J=1}^{Q} \left[\phi_M(J) - \bar{\phi}_M(J)\right]^2 \triangleq \sigma_M^2(J) \qquad (A\text{-}6)$$

The S/N ratio per se does not come directly from these three values, but σ_x^2 and σ_n^2 can be estimated separately and then the ratio formed. It has been suggested that σ_x^2 can be measured by using only the portions of the reference map that correspond to the sensor map. However, when we consider that with "live" imagery both maps have noise superimposed on them, it becomes apparent that this is not a satisfactory approach. Our approach is to (1) estimate σ_x^2 from Eq. (A-1), i.e., $\hat{\sigma}_x^2 = \phi_M(0)$; (2) with this estimate of σ_x^2 solve Eq. (A-4) for an estimate of σ_n^2; and finally (3) form the ratio of these two quantities to obtain the desired estimate. The result is

$$\hat{S/N} = \hat{\sigma}_x^2/\hat{\sigma}_n^2 = \frac{1}{\dfrac{N\,\sigma_M^2(J)}{\phi_M^2(0)} - 1} \qquad (A\text{-}7)$$

The variance of this estimate can be approximated in terms of the variance of $\sigma(J)$ and $\phi(0)$ by using a Taylor series expansion of Eq. (A-7), dropping terms higher than first order and forming the expected

*For the Product algorithm, this quantity is not measured but assumed by the statistical model to be zero.

value* of the square of the change in the S/N ratio. The interesting general formula is that

$$\text{Var } F(X, Y) = \left(\frac{\partial F}{\partial X}\right)^2 \text{Var } X + \left(\frac{\partial F}{\partial Y}\right)^2 \text{Var } Y \qquad \text{(A-8)}$$

which in this case, i.e., for F = Eq. (A-7), yields the result

$$\text{Var } (\hat{S/N}) = \frac{4N^2}{\left[\frac{N\,\sigma^2(J)}{\phi^2(0)} - 1\right]^4} \left[\frac{\sigma(J)}{\phi^2(0)}\right] \left[\text{Var } \hat{\sigma}(J) + \frac{\sigma^2(J)}{\phi^2(0)} \text{Var } \hat{\phi}(0)\right] \qquad \text{(A-9)}$$

The variance of $\hat{\phi}(0)$ is simply $\sigma^2(0)$, which can be obtained from Eq. (A-2). Obtaining the variance of $\hat{\sigma}(J)$ is somewhat more complicated, but for large values of Q ($Q \geq 30$), a close approximation† (attributed to W. K. Chow of Rand) is

---

*The expected values of $\Delta\phi(0)$, $\Delta\sigma(J)$ are taken to be zero and, with $\phi(0)$ independent of $\sigma(J)$, the expected value of the product $\Delta\phi(0)\ \Delta\sigma(J)$ is also zero.

†With $\sigma^2(J)$ given by Eqs. (A-5) and (A-6),

$$\hat{\sigma}^2(J) = \frac{1}{Q} \sum_{I=1}^{Q} \left[\phi(I) - \frac{1}{Q} \sum_{K=1}^{Q} \phi(K)\right]^2$$

or

$$\hat{\sigma}^2(J) = \frac{1}{Q} \sum_{I=1}^{Q} \left[\phi(I) - \bar{\phi}(K)\right]^2$$

where $\bar{\phi}(K)$ represents the mean value of the out-of-register correlation function (taken to be zero for the Product algorithm) and $\phi(I)$ is assumed to be distributed $N[0, \sigma^2(J)]$. $\hat{\sigma}^2(J)$ is then chi-square distributed with Q-1 degrees of freedom. Thus,

$$\sum_{I=1}^{Q} \frac{[\phi(I) - \bar{\phi}(K)]^2}{\sigma^2(J)} \sim \chi^2_{Q-1}$$

where $\sigma^2(J)$ is needed in the denominator to give the $\phi(I)$ variables a unit variance. Given that

$$\text{Var } \hat{\sigma}(J) = \frac{\sigma^2(J)}{2Q} \qquad \text{(A-10)}$$

Equation (A-9) can now be written as

$$\text{Var } (\hat{S/N}) = \frac{4N^2}{\left[\frac{\sigma^2(J)\ N}{\phi^2(0)} - 1\right]^4} \left[\frac{\sigma(J)}{\phi^2(0)}\right]^2 \left[\frac{\sigma^2(J)}{2Q} + \frac{\sigma^2(J)}{\phi^2(0)}\ \sigma^2(0)\right]$$

Expressing this in terms of σ_x^2 and σ_n^2 results in

$$\text{Var } \hat{S/N} = 4\left(\frac{\sigma_x^4}{\sigma_n^4}\right)\left(\frac{\sigma_x^2}{\sigma_n^2} + 1\right)^2\left(\frac{0.5}{Q} + \frac{2 + \sigma_n^2/\sigma_x^2}{N}\right) \qquad \text{(A-11)}$$

or

$$\text{Var } (\hat{S/N}) = 4(S/N)^2\left[1 + (S/N)\right]^2\left[\frac{0.5}{Q} + \frac{2 + 1/(S/N)}{N}\right] \qquad \text{(A-12)}$$

$$\text{Var } \sqrt{2\chi^2_{Q-1}} = 1 - \frac{1}{4(Q-1)} - \frac{1}{8(Q-1)^2} + \text{higher order terms,}$$

the Var $\sqrt{\hat{\sigma}^2(J)}$ is found to be approximately

$$\text{Var } \hat{\sigma}(J) = \text{Var } \sqrt{\frac{\sigma^2(J)}{2Q} \cdot 2\chi^2_{Q-1}}$$

or

$$\text{Var } \hat{\sigma}(J) = \frac{\sigma^2(J)}{2Q}\left[1 - \frac{1}{4(Q-1)} - \frac{1}{8(Q-1)^2}\right]$$

For reasonably large values of Q, this reduces to

$$\text{Var } \hat{\sigma}(J) = \frac{\sigma^2(J)}{2Q}$$

Table A-1 shows some values of the variance of the estimate for the range of S/N ratios of interest. As seen in this table, for S/N ratios above 1, the estimator has too large a variance to be of any practical use.

Table A-1

VARIANCE OF THE ESTIMATE OF THE S/N RATIO USING THE PRODUCT ALGORITHM FOR Q = 250 and N = 100

Actual S/N	Variance $\widehat{S/N}$	Standard Deviation $\widehat{S/N}$
0.1	0.0059	0.077
1	0.512	0.72
5	86.4	9.30
10	1113	33.4

ESTIMATING THE S/N RATIO USING MAD ALGORITHM DATA

For the MAD algorithm, the following relationships statistically hold true:

$$E\{\phi(0)\} = \sqrt{2/\pi}\,\sigma_n \tag{A-13}$$

$$\sigma^2(0) = (1 - 2/\pi)\,\sigma_n^2/N \tag{A-14}$$

$$E\{\phi(J)\} = \sqrt{2/\pi}\left(2\sigma_x^2 + \sigma_n^2\right)^{1/2} \tag{A-15}$$

$$\sigma^2(J) = (1 - 2/\pi)\,\frac{\left(2\sigma_x^2 + \sigma_n^2\right)}{N} \tag{A-16}$$

Estimates of $E\{\phi(0)\}$, $E\{\phi(J)\}$, and $\sigma^2(J)$ can be obtained from the data as described in the previous subsection of this appendix. Again, there is no way to directly estimate the S/N ratio, but it must be obtained

by estimating σ_x^2 and σ_n^2 separately and then taking the ratio. The estimate of σ_n^2 is given by

$$\hat{\sigma}_n^2 = \frac{\pi}{2} \phi_M^2(0) \qquad \text{(A-17)}$$

Table A-2 shows some values of the variance of the estimate for the same S/N ratios as are shown in Table A-1. As can be seen, for S/N ratios around 0.1, the Product algorithm provides a better estimate of the S/N ratio; but for the S/N ratios in the range of general interest (0.5 to 5), the MAD algorithm provides the better estimate. The standard deviations of the estimates appear to be small enough (from Table A-2) that this technique can be considered a useful means of estimating the S/N ratio.

Table A-2

VARIANCE OF THE ESTIMATE OF THE S/N RATIO USING THE MAD ALGORITHM FOR Q = 250 AND N = 100

Actual S/N	Variance $\hat{S/N}$	Standard Deviation $\hat{S/N}$
0.1	0.0164	0.128
1.0	0.1026	0.32
5	1.38	1.17
10	5.03	2.24

ESTIMATING THE S/N RATIO USING BOTH MAD AND PRODUCT ALGORITHM DATA

If both algorithms are used in the correlation process, a simple estimation for the S/N ratio would be

$$\hat{S/N} = \hat{\sigma}_x^2/\hat{\sigma}_n^2 = \frac{\left[\phi(0)\right]_{\text{Product}}}{\frac{\pi}{2}\left[\phi^2(0)\right]_{\text{MAD}}} \qquad \text{(A-18)}$$

The variance of this estimator is

$$\text{Var } \hat{S/N} = \left(\frac{1}{N}\right) (S/N) \; [2(\pi-1)(S/N) + 1] \qquad \text{(A-19)}$$

Table A-3 shows that the variance of this estimator is similar to that of the MAD estimator, but is somewhat better in all cases. Equation

Table A-3

VARIANCE OF THE ESTIMATE OF THE S/N RATIO USING BOTH THE MAD AND PRODUCT ALGORITHM DATA FOR Q = 250 AND N = 100

Actual S/N	Variance $\hat{S/N}$	Standard Deviation $\hat{S/N}$
0.1	0.0014	0.037
1	0.053	0.23
5	1.11	1.05
10	4.37	2.09

(A-16) cannot be used since, as discussed in the main text, N does not represent the number of independent samples. Nevertheless, an estimate of $\sigma_{\hat{S}}^2$ can be obtainted using Eqs. (A-15) and (A-17), with the result being:

$$\hat{\sigma}^2 = \left(\frac{1}{2}\right)\left(\frac{\pi}{2}\right)\left[\bar{\phi}_M(J) - \phi_M^2(0)\right]$$

where $\bar{\phi}_M(J)$ is defined in Eq. (A-5).

It follows that

$$\hat{S/N} = \frac{\hat{\sigma}_x^2}{\hat{\sigma}_n^2} = \frac{1}{2}\left[\frac{\bar{\phi}_M(J)}{\phi_M^2(0)} - 1\right] \qquad \text{(A-20)}$$

The variance of this estimate can be found in a manner similar to that described for the estimate obtained using the Product algorithm. The result is

$$\text{Var } (\hat{S/N}) = \left[\frac{\bar{\phi}(J)}{\phi^2(0)}\right]^2 \left[\text{Var } \bar{\phi}(J) + \left[\frac{\bar{\phi}(J)}{\phi(0)}\right]^2 \text{Var } \phi(0)\right] \qquad \text{(A-21)}$$

which equals

$$\text{Var } (\hat{S/N}) = \left[\frac{\bar{\phi}(J)}{\phi^2(0)}\right]^2 \left[\sigma^2(J) + \left[\frac{\bar{\phi}(J)}{\phi(0)}\right]^2 \sigma^2(0)\right] \qquad \text{(A-22)}$$

In terms of σ_x^2 and σ_n^2, this can be written as

$$\text{Var } (\hat{S/N}) = \left(\frac{\pi}{2} - 1\right)\left(2\frac{\sigma_x^2}{\sigma_n^2} + 1\right)^2\left(\frac{2}{N}\right) \qquad \text{(A-23)}$$

or

$$\text{Var } (\hat{S/N}) = \frac{1.14}{N}\left[2(S/N) + 1\right]^2 \qquad \text{(A-24)}$$

This estimator appears to yield reasonably good, unbiased estimates of the S/N ratio and correct estimates of its variance, even when spatially correlated data are used.

REFERENCES

1. Bailey, H. H., F. W. Blackwell, C. L. Lowery, and J. A. Ratkovic, *Image Correlation, Part I: Simulation and Analysis*, The Rand Corporation, R-2057/1-PR, November 1976.

2. Wessely, H. W., *Image Correlation, Part II: Theoretical Basis*, The Rand Corporation, R-2057/2-PR, November 1976.

3. Wainstein, L., and V. Zubakov, *Extraction of Signals from Noise*, Prentice Hall, Inc., New York, 1962, pp. 292-294.

4. Johnson, M. W., *Analytical Development and Test Results of Acquisition Probability for Terrain Correlation Devices Used in Navigation Systems*, AIAA Paper 72-122, presented at the Tenth Aerospace Sciences Meeting (San Diego), January 1972.

5. Hueckel, M. H., "An Operator Which Locates Edges in Digitized Pictures," *Assoc. Comp. Mach.*, Vol. 18, No. 1, January 1971, pp. 113-125.